The Report Committee for Shujuan Feng
Certifies that this is the approved version of the following report:

Mixed-Effect Modeling of Codon Usage

APPROVED BY
SUPERVISING COMMITTEE:

Supervisor:

Claus O. Wilke

Chandler W. Stolp

Mixed-Effect Modeling of Codon Usage

by

Shujuan Feng, B.S, M.S.

Report

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

Master of Science in Statistics

THE UNIVERSITY OF TEXAS AT AUSTIN

December 2010

Dedicated to my family.

Acknowledgments

First, I would like to express my sincerest gratitude to my advisor Dr. Claus Wilke for all his sound and continuous guidance, patient and friendly instructions. Without his help, this report would be impossible.

I am very grateful to Dr. Chandler Stolp for his review. I have learned a lot from his valued suggestions, comments and advice. Thanks so much!

I would like to thank Dr. Mary Parker and Martha Smith from the Department of Mathematics, and Dr. Matt Hersh from the Division of Statistics, for their substantial helps during my study in statistics.

I would also like to thank all my friends who keep encouraging me and helping me.

I would like to thank my parents and my parents-in-law for their endless love and support. Last but not least, I would like to thank my husband, Wurong, for his love, patience, support and understandings.

Mixed-Effect Modeling of Codon Usage

Shujuan Feng, M.S.Stat.
The University of Texas at Austin, 2010

Supervisor: Wilke, Claus O

Logistic mixed effects models are used to determine whether optimal codons associate with two specific properties of the expressed protein: solvent accessibility, aggregation propensity, or evolutionary conservation. Both random components and fixed structures in the models are decided by following certain selection procedures. More models are also developed by considering different factor combinations using the same selection procedure. The results show that evolutionary conservation is the most important factor for predicting for the optimal codon usage for most amino acids; aggregation propensity is also an important factor, and solvent accessibility is the least important factor for most amino acids. The results of this analysis are consistent with the previous literature, provide more straightforward way to study the research question and also more information for the insight relationships.

Table of Contents

List of Tables

Chapter 1

Introduction

1.1 General Biological Context

Codon usage bias refers to differences in the frequency of occurrence of synonymous codons in genomic DNA. A codon is a series of three nucleotides (triplets) that encodes a specific amino acid residue in a polypeptide chain. Because there are four nucleotides in DNA, adenine (A), guanine (G), cytosine (C) and thymine (T), there are 64 possible triplets encoding 20 amino acids, and three translation termination (nonsense) codons. The 20 amino acids that commonly occur in proteins are encoded by 61 different codons. This redundancy in the genetic code means that several "synonymous" codons may encode the same amino acid. We might think that mutational changes affecting these codons would not be subject to natural selection since the encoded protein sequence would be unaffected by such changes. However, this simple assumption goes against a large body of accumulated indirect molecular evidence. Different organisms often show particular preferences for one of the several codons that encode the same amino acid.

How these selection preferences arise is a much debated area of molecular evolution. Different factors have been proposed to explain codon usage

bias and the list has continued to grow. However, it is generally acknowledged that codon preferences reflect a balance between mutational biases and natural selection for translational optimization.

Selection for translational optimization, which is also referred to translational efficiency, may reflect selection for rapid translation which is also called speed selection, selection for translation with high fidelity which is also called accuracy selection, or both. Translation is an error-prone process [1] . Translation errors occur at frequencies of several misincorporations per 10,000 codons translated; precise error rates vary over nearly an order of magnitude among codons [2]. Selection for correct protein structure and function should cause codons with reduced error rates to be used more frequently at sites at which translation errors would be particularly disruptive. This selection pressure is called selection for translational accuracy [3].

To identify a signal of accuracy selection in a genome, how disruptive translation errors are at specific sites needed be measured. Evolutionary conservation [3–5] which measures the degree of evolutionary conservation by certain alignment method, can be used. The sequences of a given protein can be compared by using multiple sequence alignment (MSA) to give entropy reflecting evolutionary conservation. Differences between sequences most often represent mutations that were allowed by evolution to persist because they were harmless. Mutations occur spontaneously in each generation, randomly changing the amino acid sequences of proteins. Individuals with mutations that impair critical functions of proteins may have resulting problems that

make them less able to reproduce. Harmful mutations are lost from the gene pool because the individuals carrying them reproduce less effectively. Over time, only harmless or very rare beneficial mutations are maintained in the gene pool. This is basic rule of evolution. Where the sequences are identical, we say that sequence was conserved. Amino acids that are conserved are those most critical to the function of the protein. Thus, evolutionary conservation is also related with protein's functions. So the smaller the entropy, the more conservative the site is, and the more important that the site is.

By testing for an association between codon usage and evolutionary conservation, Akashi suggested that selection for translational accuracy should lead to inhomogeneous codon usage within genes [3]. More important sites that are less robust to translation errors and more conservative should be more frequently encoded by codons with high fidelity than other sites. Such evidence for translational accuracy selection was found in Drosophila [3] and similar results were found in Escherichia coli, yeast, worm, and mammals [4, 5].

1.2 Related Work

1.2.1 Previous Studies

In addition to linking codon usage bias to conserved (functional) or variable sites, Wilke's group linked codon usage bias to sites with specific biochemical properties. Drummond and Wilke [1] proposed a hypothesis that translational accuracy selection minimizes the misfolding of mistranslated proteins. The same group also studied whether translationally optimal codons are

associated with structurally sensitive sites, that is, sites at which translation errors are particularly likely to cause misfolding. Two structural features studied were solvent accessibility and aggregation propensity.

Residue solvent accessibilities for proteins can reflect whether the site is buried or exposed. The smaller magnitude of solvent accessibility shows the site is buried and buried residuals tend to be required for protein stability and be more important. Zhou et al.[6] considered solvent accessibility upon mutation as measures of a site's sensitivity to translation errors, and found that translationally optimal codons associate both with buried residues and with residues that are required for protein stability in E. coli, yeast, fly, and mouse. This finding provides more evidence for the hypothesis that translational accuracy selection minimizes the misfolding of mistranslated proteins, and thus tends to avoid protein aggregation [5].

Protein aggregation is the aggregation of mis-folded proteins, and is thought to be responsible for many degenerative diseases, such as Alzheimer's. It has also been implicated in CAG repeat diseases. Because protein aggregation tends to incur fitness costs, it is reasonable to suppose that the amino-acid sequence of a gene should be under selection pressure to minimize aggregation [7–10]. Wilke's group [11] investigated whether translationally optimal codons are associated with aggregation-prone sites which are particularly likely to be involved in protein-protein aggregation.

1.2.2 Data and Variables

Wilkes' group obtained genomic sequences from the following sources [6]: the Comprehensive Microbial Resource (`http://cmr.tigr.org/`) for *E. coli*, the Saccharomyces Genome Database (`ftp://genome-ftp.stanford.edu/`) for *S. cerevisiae*, the Eisen Lab (`http://rana.lbl.gov/drosophila/`) for *D. melanogaster*.

Translationally optimal codons were defined as those that are overrepresented in highly expressed genes than in gene with low expression level, and specifically, defined as the odds ratio of codon usage between highly and lowly expressed groups, calculated separately for each codon [6] :

$$C_{opt} = \frac{n_{high}/(N_{high} - n_{high})}{n_{low}/(N_{low} - n_{low})}.$$

where, n_{high} and n_{low} are the observed numbers of the codon in the highly and lowly expressed groups, and N_{high} and N_{low} are the observed numbers of the corresponding amino acid in the highly and lowly expressed groups, respectively. C_{opt} is a continuous variable.

A codon is defined as "optimal" if it showed a statistically significant increase in frequency in the highly expressed group, as determined by a chi-square test. So here a yes/no categorical variable *opt* is introduced.

The concept of entropy is used to measure the evolutionary conservation. Evolutionarily conserved sites were designated as all sites at which the amino acid was unchanged compared with the relative orthologous gene in a closely related species. Entropy in this context is a continuous variable ranging

from 0 to 2.3360. If all amino acids are the same at those sites, the entropy is 0 and it means there is no change over time and very conservative, and that site is very important. The maximum is 2.3360, corresponding to the site with most varied amino acids. The entropy is represented by a variable *ent*.

Residue solvent accessibilities for proteins with known 3D structure were obtained. After the gene sequence was aligned with the sequence from the crystal structure with MUSCLE [12], the percent solvent-accessible surface area for each aligned residue with the DSSP program [13] calculated, and the results were normalized by the reference surface areas of an extended Gly-X-Gly peptide [14]. The variable is denoted as *acc*, and is a continuous variable normally ranging from 0 to 1 with some rare values larger than 1.

Residue aggregation propensities can be predicated by the Zyggregator method based on several intrinsic properties of amino-acid sequences, including amino acid scales for secondary structure. It is defined as Z-scores and the intrinsic Z-score for aggregation, $Zagg$, and enables comparisons to be made between the aggregation propensities of different polypeptide sequences. The variable *agg* is continuous. If $Zagg$ is > 0, the sequence is more prone to aggregation than a randomly generated one; while it is less prone if $Zagg < 0$.

1.2.3 Previous Method and Results

In the previous studies [1, 6, 11], variables of entropy, residue solvent accessibility and aggregation propensity are generally classified into relative categorical variables by choosing certain cut-values. That is, they were reduced

to variables representing conservative/non-conservative, buried/exposed, and aggregation/non-aggregation, respectively. The data were first stratified by gene and synonymous codon family within each gene and constructed a separate 2×2 contingency table for each stratum. A typical example is shown in Table 1.1 from the previous paper by Zhou et al. [6].

Table 1.1: Example of a 2×2 contingency table for amino acid Thr in one particular gene in *E. coli*

	codon	Buried sites	Exposed sites
optimal	ACT, ACC	16	6
non-optimal	ACA, ACG	2	5

Note.—Codons ACT and ACC are optimal codons for amino acid Thr in *E. coli*. The odds ratio of optimal codon usage between buried and exposed sites is $(16/6)/(2/5) = 6.67$ for this contingency table.

Then either the tables for all genes and a given codon family or the tables for all genes and all codon families were combined into an overall analysis, using the Mantel-Haenszel procedure [15, 16]. In the analyses of individual amino acids, multiple testing were corrected by using the false-discovery-rate method of Benjamini and Hochberg [17].

For all amino acids excluding Met and Trp, the relationship between codon optimality and the tendency of the same codons to be preferentially used at evolutionary conservation, the relationship between codon optimality and their use at buried or exposed sites by classifying the solvent accessibility, and

also the relationship between codon optimality and their use at aggregation-prone sites were studied respectively [6].

The association between optimal codons and evolutionarily conserved sites showed that optimal codons were more preferred at conservative sites for most amino acids [1]. For the relationship between optimal codons and buried sites (solvent accessibility), it was found that optimal codons tend to be associated with buried sites(lower solvent accessibility) [6].

Not all have this same relationship, but it was present in most codon families in at least one organism. For the relationship between optimal codons and aggregation propensities, a significant preference for optimal codons at aggregation-prone residues was found in most amino acids in at least one species [11].

Further relationships were also explored. For example, the relationship between optimal codons and aggregation propensities was investigated within only exposed sites and only buried sites separately; the odds ratio of optimal codon usage between exposed-aggregation-prone and buried-non-aggregation-prone sites was studied; an association between optimal codons and buried sites was tested by considering only evolutionarily conserved residues in each species to control for evolutionary conservation. These further studies gave more information about the relationship among them.

1.3 Application of Generalized Linear Mixed Models (GLMMS) to Codon Usage Analysis

Although continuous variables for evolutionarily conservative, solvent accessibility and aggregation propensity could be used, most of the statistical analysis was done by reducing them to categorical data. Also, different cut-values might lead to different results. For example, the propensity for residue aggregation was considered aggregation-prone if $Zagg > 1$; otherwise it was defined as non-aggregation-prone. Sites whose Zagg values are close to 1 may not be well defined. Several cut-values can be used. For example, it was defined as buried site if its solvent accessibility is less 5%, 15%, or 35%. This report is a follow-up analysis of the relationships between optimal codon-usage and the same three factors: evolutionarily conservative, solvent accessibility and aggregation propensity.

By putting all data together, the research objective is to obtain further insights into whether optimal codons associate with specific properties of the expressed protein, such as its structure which is measured by solvent accessibility and aggregation propensity, or its evolutionary conservation measured by entropy. The data format is shown in Table 1.2, where one can see the data set is very big and complex with lots of missing data.

Subset data can be obtained for each amino acid, and each amino acid would be analyzed separately. Two examples (there are 18 amino acids) are shown in Table 1.3 and Table 1.4.

Regression would be an appropriate statistical analysis tool in this sit-

Table 1.2: Data Structure Display for the Whole Data in *E. coli*

N	*gene(orf)*	*acc*	*ent*	*agg*	*AA*	*codon*	C_{opt}	*opt*
1	*b*0002	0.1374	0	0.48	Met	ATG	*NA*	*NA*
2	*b*0002	0.01875	0	0	Arg	CGA	0.276	*0*
3	*b*0002	0	0	0.56	Val	GTG	0.7069	*0*
4	*b*0002	0.0082	0	0.67	Leu	TTG	0.5267	*0*
5	*b*0002	0.09	0	0.61	Lys	AAG	0.8807	*0*
...								
820	*b*0002		0	0.57	Val	GTC	0.5786	*0*
821	*b*0003	0.4426	0.8457	0.96	Met	ATG	*NA*	*NA*
822	*b*0003	0.2111	0.8457	1.58	Val	GTT	1.6676	*1*
823	*b*0003	0.2324	0	1.2	Lys	AAA	1.1355	*0*
...								
1313131	*b*4402			2.14	Leu	TTA	0.3005	*0*
1313132	*b*4402			2.13	Thr	ACA	0.2931	*0*
1313133	*b*4402			2.11	Ala	GCA	0.9856	*0*
1313134	*b*4402			1.6	Thr	ACA	0.2931	*0*

Note — N: the number of observations. *gene*: there are 4401 gene in *E. coli*, labeled as b0002 to b4402, and there are hundreds of sites (i.e. amino acids) in each gene, e.g. there are 820 sites in a gene labeled as b0002. The variable of gene is the root of the random terms in the models and is denoted as *orf* in the data and the R programs. *acc*: solvent accessibility. *ent*: evolutionarily conservative. *agg*: aggregation propensity. *AA*: amino acid. *codon*: three nucleotides code for each amino acid. C_{opt}: odds ratio of codon usage between highly and lowly expressed groups for the corresponding codon and a value of "NA" means there is no need to define a optimal codon. *opt*: a value of "1" means the codon used at this site is "optimal", "0" means the codon used at this site is "non-optimal" and a value of "NA" means there is no need to define a optimal codon. The variable *opt* is the binary response variable.

Amino acid *Met* is the beginning site for every gene and is coded by ATG, which also is the only codon for this amino acid. Therefore there is no "optimal" or "non-optimal" codon for amino acid Met.

Table 1.3: Subset Data Structure Display for Amino Acid *Arg* in *E. coli*

n	*N*	*gene(orf)*	*acc*	*ent*	*agg*	*codon*	C_{opt}	*opt*
1	2	*b*0002	0.01875	0	0	*CGA*	0.276	0
2	16	*b*0002	0.00884	0	−0.71	*CGT*	2.9336	1
3	19	*b*0002	0.3029	0.8456	0.14	*CGT*	2.9336	1
4	29	*b*0002	0.4367	0.8456	−0.19	*AGG*	0.1529	0
5	69	*b*0002		0.9979	−0.61	*CGT*	2.9336	1
6	125	*b*0002	0.0309	0	0.29	*CGT*	2.9336	1
...								
72762	1313081	*b*4401		0	1.2	*CGC*	1.1895	1
72763	1313093	*b*4402			1.36	*CGT*	2.9336	1

Note — n : the new case number. N: the observation number from the original whole data. Amino acid *Arg* can be coded by CGT, CGC, CGA, CGG, AGA or AGG. CGT and CGC are optimal codons in species *E. coli* (they can be different in different species) according to the definition stated in the early section.

Table 1.4: Subset Data Structure Display for Amino Acid *Val* in *E. coli*

n	*N*	*gene(orf)*	*acc*	*ent*	*agg*	*codon*	C_{opt}	*opt*
1	3	*b*0002	0	0	0.56	*GTG*	0.7069	0
2	11	*b*0002	0	0	1.18	*GTG*	0.7069	0
3	20	*b*0002	0	0	0.92	*GTT*	1.6676	1
4	33	*b*0002	0	0	1.16	*GTG*	0.7069	0
5	36	*b*0002	0	0	1.15	*GTC*	0.5786	0
6	48	*b*0002	0.0688	0	1.09	*GTG*	0.7069	0
...								
93285	1313121	*b*4402			−0.15	*GTG*	0.7069	0
93286	1313126	*b*4402			1.55	*GTC*	0.5786	0

Note — n : the new case number. N: the observation number from the original whole data. Amino acid *Val* can be coded by GTT, GTC, GTA or GTG. GTA and GTA are optimal codons in species *E. coli* (they can be different in different species) according to the definition stated in the early section.

uation. The response variable would be "whether the optimal codon is used", a binary category. Therefore, logistic regression would be a natural choice for a binary response variable. The predictors of interest would be evolutionarily conservative, solvent accessibility and aggregation propensity. Logistic regression models can accommodate the three original continuous prediction variables. Up to this point, a generalized linear model (GLM) seems reasonable since the three predictors are all continuous. However, the observations (i.e. the sites or amino acids) among each gene are correlated and this fact must be addressed. To treat gene type as a random effect is one of solutions.

Mixed-effects models or, mixed models for short are statistical models that incorporate both fixed-effects parameters and random effects and are used in many different disciplines. Like other regression models, they describe a relationship between a response variable and a set of covariates that have been measured or observed along with the response. The difference is that, in mixed-effects models, at least one of the covariates is a categorical variable representing experimental or observational "units" in the data set. Such units can be human or animal subjects in the study, plots of land or specific plants being studied. These units are a set of discrete levels and can be designated by numbers, but the numbers are just labels. A fixed-effects model specification is used if the set of possible levels of the covariate is fixed, exhaustive and reproducible, while random effects should be incorporated in the model if the levels observed represent a random sample from the set of all possible levels. Generalized linear mixed models (GLMMs), which are powerful but challeng-

ing tools, provide a more flexible approach for the analysis of nonnormal data with random effects and for the analysis of balanced and unbalanced grouped data and would be an appropriate choice here.

Chapter 2

Basic Theory of Generalized Linear Mixed Models (GLMMs)

2.1 Definition of GLMM

A normal linear model is defined as :

$$y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \cdots + \beta_p x_{pi} + \varepsilon_i$$

$$\varepsilon_i \sim NID(0, \sigma^2)$$

where $\beta_1, \beta_2, \cdots, \beta_p$ are the parameters of the model (regression coefficients); the error term ε_i, is the only random effect; σ^2 is the error variance.

The standard normal model can be written in a matrix form:

$$\mathbf{y} = \mathbf{X}\beta + \varepsilon$$

$$\varepsilon \sim \mathbf{N}_n(\mathbf{0}, \sigma^2 \mathbf{I}_n)$$

where $\mathbf{y} = (y_1, y_2, \cdots, y_n)'$ is the response vector; $\mathbf{X}$ is the model matrix with typical row $\mathbf{x}_i' = (x_{1i}, x_{2i}, \cdots, x_{pi})$; $\beta = (\beta_1, \beta_2, \cdots, \beta_p)'$ is the vector of regression coefficients; $\varepsilon = (\varepsilon_1, \varepsilon_2, \cdots, \varepsilon_n)'$ is the vector of errors; $\mathbf{N}_n$ represents the n-variable multivariate-normal distribution; $\mathbf{0}$ is an $n \times 1$ vector of zeros; and $\mathbf{I}_n$ is the order-n identity matrix.

The normal linear model can be estimated using least square analysis, and estimates of the unknown β are determined by minimizing a sum of squared error function.

John Nelder and Robert Wedderburn [18] formulated generalized linear model (GLM), which is a flexible generalization of ordinary least squares regression and generalizes linear regression for non-normal data. GLMs are flexible enough to include a wide range of common situations, but at the same time allow most of the familiar ideas of normal linear regression to carry over by allowing the linear model to be related to the response variable via a link function, and by allowing the magnitude of the variance of each measurement to be a function of its predicted value. In a GLM, each outcome of the dependent variables, $\mathbf{Y}$, is assumed to be generated from a particular distribution in the exponential family, a large range of probability distributions that include the normal, binomial and Poisson distributions, among others. The mean, μ, of the distribution depends on the independent variables, $\mathbf{X}$, through:

$$\mu = \mathbf{E(Y)} = g^{-1}(\mathbf{X}\beta)$$

where $\mathbf{E(Y)}$ is the expected value of $\mathbf{Y}$; $\mathbf{X}\beta$ is the linear predictor, typically denoted by the η, β is a linear combination of the unknown parameters, and g is the link function.

In this framework, the variance is typically a function, $\boldsymbol{V}$, of the mean:

$$Var(\mathbf{Y}) = V(\mu) = V(g^{-1}(\mathbf{X}\beta))$$

It is convenient if $\boldsymbol{V}$ follows from the exponential family distribution. The unknown parameters, $\boldsymbol{\beta}$, are typically estimated with maximum likelihood, maximum quasi-likelihood, or Bayesian techniques.

A GLM is defined by specifying two components. The response should be a member of the exponential family distribution and the link function describes how the mean of the response and a linear combination of the predictors are related. The link function provides the relationship between the linear predictor and the mean of the distribution. There are many commonly used link functions and certain natural choices called canonical are preferred, although their choice can be somewhat arbitrary. When using a distribution function with a canonical parameter θ, a link function exists which allows for $\mathbf{X}^T\mathbf{Y}$ to be a sufficient statistic for β. This occurs when the link function equates θ and the linear predictor. Table 2.1 shows some canonical link functions.

Table 2.1: Canonical Link Functions

Distribution	Name	Link
Normal	Identity	$\eta = \mu$
Exponential	Inverse	$\eta = \mu^{-1}$
Gamma		
Inverse Gaussian	Inverse squared	$\eta = \mu^{-2}$
Poisson	Log	$\eta = \ln \mu$
Binomial	Logit	$\eta = \ln\left(\frac{\mu}{(1-\mu)}\right)$
Multinomial		

Linear mixed-effect models which include additional random-effect terms are an extension to the generalized linear model and they are particularly use-

ful in settings where repeated measurements are made on the same statistical units, or where measurements are made on clusters of related statistical units.

Linear mixed model can be expressed in the following forms:

$$y_{ij} = \beta_1 x_{1ij} + \cdots + \beta_p x_{pij} + b_{i1} z_{1ij} + \cdots + b_{iq} z_{qij} + \varepsilon_{ij}$$

$$b_{ik} \sim N(0, \psi_k^2), Cov(b_k, b_{k'}) = \psi_{kk'}$$

$$\varepsilon_{ij} \sim N(0, \sigma^2 \lambda_{ijj}, Cov(\varepsilon_{ij}, \varepsilon_{ij'}) = \sigma^2 \lambda_{ijj'}$$

where

- y_{ij} is the value of the response variable for the jth of n_i observations in the ith of M groups or clusters;
- $\beta_1, \cdots, \beta_p$ are the fixed population coefficients to be estimated;
- $x_{1ij}, \cdots, x_{pij}$ are the regressors for observation j in group i, associated with the fixed parameters β; the first regressor for the constant term is defined as, $x_{1ij} = 1$;
- $b_{i1}, \cdots, b_{iq}$ are the random effect coefficients for group i, assumed to be multivariately normally distributed, and to vary across groups;
- $z_{1ij}, \cdots, z_{qij}$ are the regressors associated with the random effect parameters b;
- ψ_k^2 are the variances and $\psi_{kk'}$ are the covariances among the random effects. All are assumed to be constant across groups;

- ε_{ij} is the error for observation j in group i, where the errors for group i are assumed to be multivariately normally distributed; and
- $\sigma^2\lambda_{ijj'}$ are the covariances between errors in group i.

The linear mixed model can be expressed in matrix form as follows,

$$\boldsymbol{y_i} = \boldsymbol{X_i}\boldsymbol{\beta} + \boldsymbol{Z_i}\boldsymbol{b_i} + \boldsymbol{\varepsilon_i}$$

$$\boldsymbol{b_i} \sim \boldsymbol{N_q}(\mathbf{0}, \boldsymbol{\Psi})$$

$$\boldsymbol{\varepsilon_i} \sim \boldsymbol{N_{n_i}}(\mathbf{0}, \boldsymbol{\sigma^2}\boldsymbol{\Lambda_i})$$

where

- $\boldsymbol{y_i}$ is the $n_i \times 1$ response vector for observations in the ith group;
- $\boldsymbol{X_i}$ is the $n_i \times p$ matrix of explanatory variable for the observations in group i;
- $\boldsymbol{\beta}$ is the $p \times 1$ vector of the fixed coefficients;
- $\boldsymbol{Z_i}$ is the $n_i \times q$ matrix of explanatory variables associated with the random effects for the observations in group i;
- $\boldsymbol{b_i}$ is the $q \times 1$ vector of random effect coefficients for group i;
- $\boldsymbol{\varepsilon_i}$ is the $n_i \times 1$ vector of errors for observation in group i;
- $\boldsymbol{\Psi}$ is the $q \times q$ covariance matrix for the random effects; and

- $\boldsymbol{\sigma^2\Lambda_i}$ is the $n_i \times n_i$ covariance matrix for the errors in group i.

Generalized linear mixed models (GLMMs), which extend GLMs by the inclusion of random effects in the predictor by combining the properties of the above two statistical frameworks: generalized linear models (which handle nonnormal data by using link functions) and linear mixed models (which incorporate random effects).

In this report, logistic mixed effects models will be used since the response is binary and the corresponding canonical link is logistic.

2.2 Estimation of GLMMs

Maximum likelihood (ML), is generally used to estimate the parameters in a linear mixed model. For simple analyses where the response variables are normal, the design is balanced, and all random effects are nested effects, classical ANOVA methods based on computing differences of sums of squares give the same answers as ML approaches. However, this equivalence breaks down and is even computationally infeasible for more complex LMMs and GLMMs.

Various ways to approximate the likelihood to estimate GLMM parameters have been proposed, including pseudo- and penalized quasi-likelihood (PQL)[19–21], Laplace approximations [22], Gauss-Hermite quadrature (GHQ) [23], as well as Markov chain Monte Carlo (MCMC) algorithms [24]. In all of these approaches, one must distinguish between standard ML estimation,

which estimates the standard deviations of the random effects assuming that the fixed-effect estimates are precisely correct, and restricted maximum likelihood (REML) estimation, a variant that averages over some of the uncertainty in the fixed-effect parameters [25, 26]. The REML estimates of variance components are often preferred to the ML estimates because ML underestimates random-effect standard deviations, except in very large data sets.

MCMC is usually used in a Bayesian framework, which incorporates prior information based on previous knowledge. PQL is the simplest and most widely used GLMM approximation and it is implemented widely in statistical packages. However, PQL has two main disadvantages. One is that it gives biased parameter estimates if the standard deviations of the random effects are large, especially with binary data. [27, 28] As a rule of thumb, PQL works poorly for Poisson data when the mean number of counts per treatment combination is less than five, or for binary data [29]. Another disadvantage is that it computes a quasi-likelihood rather than a true likelihood.

GHQ [23] is more accurate, but it is slow and the speed of GHQ decreases rapidly with increasing numbers of random effects. Laplace approximation [22] reduces bias and approximates the true GLMM likelihood rather than a quasi-likelihood, allowing the use of likelihood-based inference.

2.3 Linear Mixed-Effects Models Using S4 Classes (lme4) Package in R

Although the theory and estimation methods of GLMMs are rather complicated, using program to estimate models makes the application of GLMMs less formidable. This report uses the function glmer in the package of linear mixed-effects models using S4 classes (lme4) by Douglas Bates and Martin Maechler (http://cran.r-project.org/web/packages/lme4/index.html) in R [30], where Laplace approximation is used since we have GLMMs for binary data.

An example for the basic form of glmer function is shown as follows :

$$glmer(formula, data, family)$$

where

- formula is a two-sided linear formula object describing the fixed-effects part of the model, with the response on the left of a "$\sim$" operator and the terms, separated by "+" operators, on the right. The vertical bar character "|" separates an expression for a model matrix and a grouping factor.
- data is the data frame containing the variables named in formula.
- family is a GLM family. If family is missing then a linear mixed model is fit; otherwise a generalized linear mixed model is fit.

The following is a real example used in the report :

$$glmer(opt \sim ent * acc * agg + (1|orf), data = aaA, family = binomial)$$

where

- opt is the binary response variable.
- ent, acc, agg are the three predictors. By $ent * acc * agg$, it means that all the interactions are included in the model. So it is equal to $acc + agg + ent + acc * agg + ent * agg + acc * ent + acc * agg * ent$.
- aaA is data set for an amino acid.
- Here a binomial GLM family is used.

After deciding estimation methods and the application of program, choosing appropriate model selection procedures is the next important step in this report.

Chapter 3

Model Selection

Although GLMMs themselves are uncontroversial, describing how to use them to analyze data necessarily raise controversial statistical issues [31] such as the validity of stepwise regression [32] and the use of Bayesian statistics [33]. A suite of model simplification techniques have been developed, and the notion of a minimum adequate model (MAM) has become common in ecology [31]. A MAM is defined as the mode that contains the minimum number of predictors that satisfy some criterion, for example, the model that only contains predictors that are significant at some pre-specified probability level. Finding such a model is not straightforward, but most statistical packages offer algorithms for model selection in multiple regression [31].

3.1 General Model Selection and Procedures for Mixed Models

Although it is strongly discouraged to automate stepwise regression with many potential predictors, certain disciplined hypothesis testing for model reduction is still considered appropriate in some situations [31, 34]. A model selection procedure generally compares fits of candidate models. It can be done either by using hypothesis tests which test simpler nested models against

more complex models [35] or by using information-theoretic approaches, such as AIC or BIC [36].

3.1.1 Model Selection Issues

Likelihood ratio tests compare the relative deviance between nested models. In most cases, such a test for random effects will involve hypotheses of the form: $H_0 : \sigma^2 = 0$. The standard deviation of the asymptotic χ^2 distribution for the likelihood ratio statistics depends on the null hypothesis lying in the interior of the parameter space. This assumption is broken when we test if a variance is 0. Therefore, the distribution under the null hypothesis in this circumstance won't be an approximate χ^2 distribution. This is called boundary effects. A numerical method is required to do precise testing. If the χ^2 distribution is used with the appropriate degrees of freedom, then the test will tend to be conservative. That is, the p value will tend to be larger than they should be so that one can be fairly confident that it is actually significant if one observes a significant random effect using the χ^2 distribution approximation. However, small but not significant p values might spur one to use more accurate, but time consuming, bootstrap methods [31]. So likelihood ratio tests for random terms are conservative, therefore increasing the risk of type II errors. Information-theoretic approaches using AIC or BIC suffer from analogous problems and can not avoid the boundary effects problem[31, 37, 38]. Despite being conservative due to the boundary effects, LR tests or AIC (BIC) are still widely used and generally appropriate for inference concerning random

factors with certain corrections to address boundary problems.

However, the LR test is not recommended for testing fixed effects in GLMMs, because it is unreliable especially with small to moderate sample sizes. Additionally, for the fixed effects, the p-values of the t test from the model summary could be strongly "anti-conservative" and therefore sometimes overstating the importance of some effects because the t distribution assumption for the fixed estimates is not true in the mixed model. Some methods, such Markov chain Monte Carlo or some bootstrap methods, are used to obtain more accurate results and they tend to give similar p values except in small samples. The usual bootstrap approach is nonparametric in that no underlying distribution is assumed. If the errors and random effects can be assumed to be normally distributed, a technique that called the parametric bootstrap can also be used. In the parametric bootstrap, the probability of observing an LRT of what we observed or greater, given that the null hypothesis is true, needs to be estimated. Under the null hypothesis, a simulation approach would generate data under this model, fit the null and alternative models and then compute the LRT. The process would be repeated a large number of times and the proportion of LRTs exceeding the observed value would be used to estimate the p value.

3.1.2 General Procedure

For a mixed model,the model selection procedure can be more complicated, even stepwise selection poses a challenge, since both random and fixed

effect components need to be considered. Zuur [39] recommended a procedure which is basically the one followed by this report. In his book, Zuur stated:

> 1: Start with a model where the fixed model component contains all explanatory variables and as many interactions as possible. This is called the "beyond optimal" model. If this is impractical, e.g. due to a large number of explanatory variables, interactions, or numerical problems, use a selection of explanatory variables that you think are most likely to contribute to the optimal model.
>
> 2: Using the beyond optimal model, fit the optimal structure of the random component. Because we have as many explanatory variables as possible in the fixed component, the random component (hopefully) doesn't contain any information that we would like to have in the fixed component. Compare random effects using the Likelihood Ratio test (REML): because the ML estimators for the variance terms are biased by ML. As well as using the (REML) LR, we can also use AIC or BIC.
>
> 3: Once the optimal random structure has been found, it is time to find the optimal fixed structure. We can either use the F-statistics or t statistic obtained with REML estimation or compare nested models. To compare models with nested fixed effects (but the same random structure), ML estimation must be used.
>
> 4. Present the final results using REML estimation.

3.2 The Model Selection Procedure and Model Specifications in this Report

The above procedure has been used in this report with some modifications.

3.2.1 A Beyond Optimal Model

First of all, a "beyond optimal" fixed model with all three interaction terms is estimated. The fixed model component in this "beyond optimal" model contains all explanatory variables and as many interactions as possible. And the random intercept by gene is added into the model by default according to the design of the study. This optimal fixed model can be expressed as:

$$\begin{aligned}\eta_{ij} &= \beta_0 + \beta_1 acc_{ij} + \beta_2 ent_{ij} + \beta_3 agg_{ij} + \beta_4 ent_{ij} agg_{ij} \\ &+ \beta_5 ent_{ij} acc_{ij} + \beta_6 acc_{ij} agg_{ij} + \beta_7 acc_{ij} agg_{ij} ent_{ij} + b_i \qquad (M_0)\end{aligned}$$

where

- $\eta_{ij} = logit(\mu) = log(\frac{\pi}{1-\pi})$,and π is the probability of an optimal codon used on a specific site;
- $\beta_1, \cdots, \beta_7$ are the coefficients for the predictors;
- *acc* is the variable of solvent accessibility ,*agg* is the variable of aggregation propensity, *ent* is the variable of evolutionary conservation measured by entropy;

- b_i is the random intercept by gene type.

After deciding the "beyond optimal model", the next step is to decide the random components.

3.2.2 Optimal Random Components

For random effects, both LR and AIC are used to test random effects. LR tests tend to underestimate the significance of the random effect and give conservative results due to the boundary issue. If the p value of LR shows a random effect is significant, we feel pretty confident that the random effect does exist; if it is very non-significant, we feel comfortable removing the random effect; but if it is a little larger than our α level, which is 0.05 in this analysis, we may have the risk of underestimating the random effect. $0.5 \times p$ is recommended to reduce this risk and has been used in this analysis. Using AIC, which can be used for comparing non-nested models, has the same issue of boundary effects and is used here to check if there is a conflict between LR and AIC results. If there is a conflict, a further checking is needed. The estimation method would be different from the Random Error Likelihood Ratio test (REML) which is not applicable since approximation LR must be used for GLMM.

Specifically, the comparisons are done as the following way:

First, choose a best model by comparing the following three models which include a random intercept and also one random slope with the "beyond

optimal model" which only has a random intercept (notice they all have the same beyond optimal fixed components).

$$\eta_{ij} = \beta_0 + \beta_1 acc_{ij} + \beta_2 ent_{ij} + \beta_3 agg_{ij} + \beta_4 ent_{ij} agg_{ij}$$

$$+ \beta_5 ent_{ij} acc_{ij} + \beta_6 acc_{ij} agg_{ij} + \beta_7 acc_{ij} agg_{ij} ent_{ij} + b_{i0} + b_{i1} acc_{ij} \qquad (M_1)$$

$$\eta_{ij} = \beta_0 + \beta_1 acc_{ij} + \beta_2 ent_{ij} + \beta_3 agg_{ij} + \beta_4 ent_{ij} agg_{ij}$$

$$+ \beta_5 ent_{ij} acc_{ij} + \beta_6 acc_{ij} agg_{ij} + \beta_7 acc_{ij} agg_{ij} ent_{ij} + b_{i0} + b_{i1} agg_{ij} \qquad (M_2)$$

$$\eta_{ij} = \beta_0 + \beta_1 acc_{ij} + \beta_2 ent_{ij} + \beta_3 agg_{ij} + \beta_4 ent_{ij} agg_{ij}$$

$$+ \beta_5 ent_{ij} acc_{ij} + \beta_6 acc_{ij} agg_{ij} + \beta_7 acc_{ij} agg_{ij} ent_{ij} + b_{i0} + b_{i1} ent_{ij} \qquad (M_3)$$

If the "beyond optimal model" M_0 which only has a random intercept is favored over other three models, the random component selection is finished and the random component is decided to be just the random intercept and no random slopes significant enough to be in the model. Otherwise, the random component selection procedure continues based on the most favored model with the random intercept and a random slope. Suppose the model M_1 is the most favored. Alternative potential modes would be the followings with one more random slope.

$$\eta_{ij} = \beta_0 + \beta_1 acc_{ij} + \beta_2 ent_{ij} + \beta_3 agg_{ij} + \beta_4 ent_{ij} agg_{ij}$$

$$+ \beta_5 ent_{ij} acc_{ij} + \beta_6 acc_{ij} agg_{ij} + \beta_7 acc_{ij} agg_{ij} ent_{ij} + b_{i0} + b_{i1} acc_{ij} + b_{i2} ent_{ij} \quad (M_4)$$

$$\eta_{ij} = \beta_0 + \beta_1 acc_{ij} + \beta_2 ent_{ij} + \beta_3 agg_{ij} + \beta_4 ent_{ij} agg_{ij}$$

$$+ \beta_5 ent_{ij} acc_{ij} + \beta_6 acc_{ij} agg_{ij} + \beta_7 acc_{ij} agg_{ij} ent_{ij} + b_{i0} + b_{i1} acc_{ij} + b_{i2} agg_{ij} \quad (M_5)$$

If it turns out Model M_1 is favored over models with two random slopes, the random component selection is finished at this step and the random component is decided to be just Model M_1 with the random intercept and a random slope. Otherwise the random component selection procedure continues based on the most favored model with the random intercept and two random slopes. Suppose the model M_4 is the most favored. Next optional mode would include all random slopes.

$$\eta_{ij} = \beta_0 + \beta_1 acc_{ij} + \beta_2 ent_{ij} + \beta_3 agg_{ij} + \beta_4 ent_{ij} agg_{ij}$$

$$+\beta_5 ent_{ij} acc_{ij} + \beta_6 acc_{ij} agg_{ij} + \beta_7 acc_{ij} agg_{ij} ent_{ij} + b_{i0} + b_{i1} agg_{ij} + b_{i2} ent_{ij} + b_{i3} acc_{ij} \quad (M_6)$$

If it turns out Model M_4 is favored over the model with all random slopes, the random component is decided to be just Model M_4 with the random intercept and two random slopes. Otherwise the random component should include all random slopes.

After the random components are decided, the next step is to decide optimal fixed structure.

3.2.3 Optimal Fixed Components

We didn't use Markov chain Monte Carlo or bootstrap methods since they are very time consuming and we have many models and very large data sets. We didn't use the p-values of the t test from the model summary either, because the distribution of the parameter estimates is not symmetric and does

not converge to a normal distribution, and the p-values could be strongly "anti-conservative". Instead, multiple testing is used for selecting and assessing the fixed effects according to the model summary. Although the multiple testing doesn't solve the above "anti-conservative" issue directly, it is certainly much more conservative than the original p-values from the model summary.

Start with the full mixed model with the optimal random structures and use multiple testing to remove the most non-significant term, and then re-run the model without that most non-significant term and continue to remove the most non-significant one until all the fixed terms in the model are significant. For the random structure, forward selection is used with a cut $p = 0.05$; For the fixed effects, backward elimination is used, with a critical p value of 0.05 remove.

After both random and fixed components are decided, final results can be presented.

3.2.4 Acknowledge Alternative Possible Models

It should be acknowledged that it is possible that the optimal model (if it exists) is never considered due to the nature of the model selection process, especially when there are strong correlations among predictors. When several variables, which are highly correlated, are each associate with the response, we have to take care that we don't conclude that the variables we drop have nothing to do with the response. Although the magnitudes of the correlation coefficients among evolutionarily conservative (entropy), solvent accessibility

and aggregation propensity are quite small and we do not see any pattern from the data plots, the correlations are significant due to the large degree of freedom and are expected conceptually.

For such a situation, normally only one model is not enough to explain data and provide enough useful information. We should acknowledge the possibility of alternative conflicting models and seek them. It is recommended to search several models or at least acknowledge their existence especially if the model purpose is to explain the data, not to predict the future. In complex data analysis involving several variables, several models could be found to fit the data well [40]. Thus, searching more models, by only considering one or two predictors separately, is also done in this report for each amino acid for the three interested species by following the same model selection procedure as discussed above. That is, seven procedures are carried on for each amino acid, by considering only *acc*, *ent*, *agg*, *acc* and *ent*, *acc* and *agg*, *agg* and *ent*, and all three factors (*acc, ent and agg*).

Three species (*E. coli*, *yeast* and fly) are analyzed in this report. 17 or 18 amino acids are analyzed in each species and seven model selection procedures are carried on for each amino acid according to the above selection procedures. Therefore, substantial useful results can be provided.

Chapter 4

Results and Discussion

4.1 Results of Analysis

During the procedure of selection of optimal random components, almost all results from LR tests are consistent with AIC tests. There are only a few cases with conflicts in which the p-values are on the edge (a little larger than 0.05). Once the rule of $0.5 \times P$ is applied, they are all consistent.[1]

All the results are displayed in Tables 4.1 - 4.21.

The estimated coefficients are translated into odds ratios and random effects are also shown in the tables. The fixed effect terms are of the main interest. We can see that similar results are obtained for the three species: *E. coli*, *yeast* and fly. With respect to solvent accessibility (*acc*), 10 of 17 amino acids show significance in *E. coli* (see Table 4.1); 11 of 18 amino acids show significance in *yeast* (see Table 4.2); 7 of 18 amino acids show significance in *fly* (see Table 4.3).

In terms of evolutionarily conservative (*ent*), 14 of 17 amino acids show significance in *E. coli* (see Table 4.4); 16 of 18 amino acids show significance

[1]The final R programs only present the selection procedure by comparing AICs.

in *yeast* (see Table 4.5); 18 of 18 amino acids show significance in *fly* (see Table 4.6).

For aggregation propensity (*agg*), 11 of 17 amino acids show significance in *E. coli* (see Table 4.7); 13 of 18 amino acids show significance in *yeast*(see Table 4.8); 16 of 18 amino acids show significance in *fly* (see Table 4.9).

For both *acc* and *ent*, 3 of 17 amino acids show significance for *acc* and 12 of 17 for *ent* in *E. coli* (see Table 4.10); 2 of 18 amino acids show significance for *acc* and 14 of 18 for *ent* in *yeast* (see Table 4.11); 1 of 18 amino acids show significance for *acc* and 18 of 18 for *ent* in *fly* (see Table 4.12).

With both *acc* and *agg* present in the model, 9 of 17 amino acids show significance for *acc* and 14 of 17 for *agg* in *E. coli* (see Table 4.13); 7 of 18 amino acids show significance for *acc* and 10 of 18 for *agg* in *yeast* (see Table 4.14); 3 of 18 amino acids show significance for *acc* and 16 of 18 for *agg* in *fly* (see Table 4.15).

With both *agg* and *ent*, 12 of 17 amino acids show significance for *agg* and 11 of 17 for *ent* in *E. coli* (see Table 4.16); 2 of 18 amino acids show significance for *agg* and 14 of 18 for *ent* in *yeast* (see Table 4.17); 14 of 18 amino acids show significance for *agg* and 16 of 18 for *ent* in *fly* (see Table 4.18);.

Finally, with all three factors *acc ent* and *agg*, 3 of 17 amino acids show significance for *acc* 11 of 17 for *agg* and 13 of 17 for *ent* in *E. coli* (see Table 4.19); 1 of 18 amino acids show significance for *acc* , 2 of 18 for *agg* and

15 of 18 for *ent* in *yeast* (see Table 4.20); 3 of 18 amino acids show significance for *acc* , 12 of 18 for *agg* and 13 of 18 for *ent* in *fly* (see Table 4.21).

4.2 Conclusion of Analysis

Interpreting models built on observational data is problematic. There are many opportunities for errors to distort results and conclusions can vary substantially [40]. First of all, what does the estimate coefficient β_1 mean? The interpretation is that a change of β_1 in the response is produced when there is one unit change in the predictor and other predictors are held constant [40]. But individual variables generally cannot change without changing others in real-world settings. Here, for example, we cannot increase one unit of *acc* while still keeping *agg* and *ent* constant. As a consequence, we pay more attention to the direction of the sign and relative values.

For most amino acids, *ent*(which measures the evolutionary conservation) shows significance in all seven procedures and the estimate coefficients are negative, which is consistent with the previous research that finds optimal codons to be preferred at conservative sites [1]. The *acc* and *agg* also show significance for most amino acids by only including one factor in the model, but lost the significance when two or three factors are included in the model. Almost all estimated coefficients for *agg* (which measures residue aggregation propensities) are positive, which is also consistent with previous research. For *acc* (which measures residue solvent accessibilities), most are negative, but some are positive. However they are also quite consistent with the previous

findings.

Generally, the results show that the factor of evolutionary conservation is the most important for optimal codon usage for most amino acids; aggregation propensity also is an important factor; solvent accessibility is the least important factor for most amino acids. However, further biological conclusions are rather complicated and far beyond this report.

This report carried out a a further study of the relationship between optimal codon usage and certain factors of the expressed protein by using GLMMs. First of all, the results of the analysis are consistent with the prior research confirming earlier findings. More importantly, this analysis by GLMMs provides a valid and more straightforward way to address the research questions. Exploring a range model specifications by considering different factor combinations provides more insight into the relationships among the factor explaining optimal codon usage.

Table 4.1: Results for *E. coli* by considering only *acc*

AA	*Fixed effects*		*Random Effects(standard deviation)*	
	coefficients	Odds Ratio	Random Intercept	Random Slope
Ala	0.1844	1.2025^{***}	0.4479	
Arg	-0.099	$0.9057^{N.S.}$	1.0332	0.4808
Asn	-0.3249	0.7226^{***}	0.7644	0.2694
Asp	0.0787	1.0819^{*}	0.4554	0.2892
Cys			0.3103	
Gln	-0.184	0.8319^{***}	0.4204	
Glu			0.3118	
Gly	-0.131	0.8772^{***}	0.59	
His	-0.0141	$0.9860^{N.S.}$	0.5759	0.2712
Ile			0.4881	
Leu	-0.2342	0.7912^{***}	0.6159	
Lys*				
Phe			0.5582	
Pro	-0.2518	0.7774^{***}	0.5948	
Ser	-0.3002	0.7407^{***}	0.5711	0.3883
Thr	-0.2845	0.7524^{***}	0.4878	
Tyr			0.4556	
Val	0.1765	1.1930^{***}	0.4762	

Note —AA: amino acid; random effects are shown as "standard deviations"
Significance levels: $^{***}P < 0.001$; $^{**}P < 0.01$;$^{*}P < 0.05$.
N.S. nonsignificant, but in the model because its random slope is in the model.
Lys*: no optimal codon.

Table 4.2: Results for *S.cerevisiae (yeast)* by considering only *acc*

AA	*Fixed effects*		*Random Effects(standard deviation)*	
	coefficients	Odds Ratio	Random Intercept	Random Slope
Ala	−0.0917	0.9124**	0.5863	
Arg	−0.1038	0.9014**	0.6376	
Asn			0.3969	
Asp	0.0949	1.0995***	0.367	
Cys	−0.1999	0.8188**	0.4042	
Gln			0.3545	
Glu			0.2994	
Gly	−0.2756	0.7591***	0.8738	0.4528
His			0.3354	
Ile	−0.2369	0.7891***	0.7011	0.201
Leu	−0.0936	0.9106**	0.4657	
Lys	0.0811	1.0845**	0.4276	0.0936
Phe	−0.0904	0.9136*	0.4129	
Pro			0.4168	
Ser	−0.1484	0.8621***	0.5296	0.0838
Thr			0.4296	
Tyr			0.4	
Val	−0.1515	0.8594***	0.5773	

Note —AA: amino acid; random effects are shown as "standard deviations"
Significance levels: $^{***}P < 0.001$; $^{**}P < 0.01$;$^{*}P < 0.05$.
N.S. nonsignificant, but in the model because its random slope is in the model.

Table 4.3: Results for *D.melanogaster (fly)*by considering only *acc*

AA	*Fixed effects*		*Random Effects(standard deviation)*	
	coefficients	Odds Ratio	Random Intercept	Random Slope
Ala	−0.151	0.8598***	0.4343	0.1417
Arg			0.4727	
Asn			0.4537	
Asp	0.0669	1.0692***	0.445	
Cys			0.4151	
Gln			0.5256	
Glu			0.5653	
Gly			0.4	
His			0.3802	
Ile	−0.0828	0.9205***	0.5165	
Leu	−0.0774	0.9255***	0.6	
Lys			0.6007	
Phe	−0.1201	0.8868***	0.5455	
Pro	−0.0232	$0.9771^{N.S.}$	0.4188	0.0858
Ser			0.459	
Thr	−0.0823	0.921***	0.4288	
Tyr			0.4425	
Val	−0.052	0.9493*	0.5063	

Note —AA: amino acid; random effects are shown as "standard deviations"
Significance levels: $^{***}P < 0.001$; $^{**}P < 0.01$;$^{*}P < 0.05$.
N.S. nonsignificant, but in the model because its random slope is in the model.

Table 4.4: Results for *E. coli* by considering only *ent*

AA	*Fixed effects*		*Random Effects(standard deviation)*	
	coefficients	Odds Ratio	Random Intercept	Random Slope
Ala	0.0608	1.0627^{*}	0.546	0.1656
Arg	−0.6106	0.5430^{***}	0.9099	0.2979
Asn	−0.3715	0.6897^{***}	0.7142	0.1663
Asp	-0.0303	$0.9702^{N.S.}$	0.4412	0.0997
Cys			0.3103	
Gln	−0.2379	0.7883^{***}	0.4197	0.1599
Glu	−0.0545	0.9470^{*}	0.2726	
Gly	−0.171	0.8428^{***}	0.6449	
His	−0.2959	0.7439^{***}	0.5878	0.1433
Ile	−0.1137	0.8925^{***}	0.5566	0.1166
Leu	−0.2277	0.7964^{***}	0.5714	0.1846
Lys*				
Phe	−0.1391	0.8701^{***}	0.5905	
Pro	−0.2237	0.7996^{***}	0.4674	
Ser	−0.3487	0.7056^{***}	0.6756	0.2572
Thr	−0.2367	0.7892^{***}	0.5636	0.1132
Tyr	-0.085	$0.9185^{N.S.}$	0.4277	0.1635
Val	−0.0004	0.9996^{***}	0.5596	0.1734

Note —AA: amino acid; random effects are shown as "standard deviations"
Significance levels: $^{***}P < 0.001$; $^{**}P < 0.01$;$^{*}P < 0.05$.
N.S. nonsignificant, but in the model because its random slope is in the model.

Table 4.5: Results for *S.cerevisiae (yeast)* by considering only *ent*

AA	*Fixed effects*		*Random Effects(standard deviation)*	
	coefficients	Odds Ratio	Random Intercept	Random Slope
Ala	−0.1835	0.8324***	0.6329	0.1672
Arg	−0.3929	0.6751***	0.641	0.3334
Asn	−0.0819	0.9214***	0.4675	0.1109
Asp			0.3247	
Cys	−0.2637	0.7682***	0.4798	0.2964
Gln	−0.0587	0.943*	0.4472	
Glu	−0.1075	0.8981***	0.3981	0.2027
Gly	−0.3481	0.706***	0.6928	0.3526
His			0.3354	
Ile	−0.3677	0.6923***	0.7366	0.2431
Leu	−0.1587	0.8533***	0.4265	0.1271
Lys	−0.1205	0.8865***	0.4788	0.1797
Phe	−0.0781	0.9249**	0.3546	
Pro	−0.088	0.9158***	0.4836	0.1317
Ser	−0.3663	0.6933***	0.5152	0.2054
Thr	−0.0644	0.9376**	0.5511	0.1629
Tyr	−0.1192	0.8876***	0.3722	
Val	−0.1663	0.8468***	0.6005	0.2071

Note —AA: amino acid; random effects are shown as "standard deviations"
Significance levels: $^{***}P < 0.001$; $^{**}P < 0.01$;$^{*}P < 0.05$.
N.S. nonsignificant, but in the model because its random slope is in the model.

Table 4.6: Results for *D.melanogaster (fly)* by considering only *ent*

AA	*Fixed effects*		*Random Effects(standard deviation)*	
	coefficients	Odds Ratio	Random Intercept	Random Slope
Ala	−0.2886	0.7493 ***	0.3259	0.1465
Arg	−0.4367	0.6462 ***	0.4067	0.1759
Asn	−0.097	0.9076 ***	0.4241	0.0605
Asp	−0.1434	0.8664 ***	0.3917	0.0738
Cys	−0.1758	0.8388 ***	0.3504	0.2165
Gln	−0.162	0.8504 ***	0.4158	0.1643
Glu	−0.2859	0.7513 ***	0.4324	0.1313
Gly	−0.1812	0.8343 ***	0.51	0.1737
His	−0.1405	0.8689 ***	0.3491	0.1386
Ile	−0.1185	0.8883 ***	0.319	
Leu	−0.1605	0.8517 ***	0.4327	0.1469
Lys	−0.2677	0.7651 ***	0.4847	0.1368
Phe	−0.2603	0.7708 ***	0.5034	0.1508
Pro	−0.228	0.7961 ***	0.3089	
Ser	−0.166	0.847 ***	0.3549	
Thr	−0.1234	0.8839 ***	0.3698	0.1328
Tyr	−0.1311	0.8771 ***	0.3843	
Val	−0.2391	0.7873 ***	0.4271	0.1798

Note —AA: amino acid; random effects are shown as "standard deviations"
Significance levels: $^{***}P < 0.001$; $^{**}P < 0.01$;$^{*}P < 0.05$.
N.S. nonsignificant, but in the model because its random slope is in the model.

Table 4.7: Results for *E. coli* by considering only *agg*

AA	*Fixed effects*		*Random Effects(standard deviation)*	
	coefficients	Odds Ratio	Random Intercept	Random Slope
Ala	0.0017	$1.0017^{N.S.}$	0.4695	0.0331
Arg	−0.0206	$0.9796^{N.S.}$	0.962	0.0957
Asn	0.0584	1.0601***	0.6698	0.05
Asp	0.0362	1.0369***	0.4038	0.0831
Cys	−0.0433	0.9576*	0.3184	
Gln	0.0409	1.0417***	0.431	
Glu			0.3118	
Gly	0.1043	1.1099***	0.606	
His	0.1058	1.1116***	0.5099	
Ile	0.0356	1.0362***	0.51	
Leu	−0.0106	$0.9894^{N.S.}$	0.6452	0.0639
Lys*				
Phe			0.5582	
Pro			0.617	
Ser	0.0649	1.0670***	0.5132	
Thr	0.1634	1.1775***	0.5032	
Tyr	0.0944	1.0990***	0.4676	
Val	−0.0953	0.9091***	0.4832	

Note —AA: amino acid; random effects are shown as "standard deviations"
Significance levels: $^{***}P < 0.001$; $^{**}P < 0.01$;$^{*}P < 0.05$.
N.S. nonsignificant, but in the model because its random slope is in the model.
Lys*: no optimal codon.

Table 4.8: Results for *S.cerevisiae (yeast)* by considering only *agg*

AA	*Fixed effects*		*Random Effects(standard deviation)*	
	coefficients	Odds Ratio	Random Intercept	Random Slope
Ala	0.0264	1.0268^{***}	0.4171	
Arg	0.0224	1.0227^{**}	0.4582	0.0604
Asn	0.0274	1.0278^{***}	0.3187	
Asp			0.3247	
Cys			0.3661	
Gln	0.018	1.0182^{*}	0.301	0.0762
Glu	0.0178	1.018^{**}	0.2596	0.0644
Gly	0.0634	1.0655^{***}	0.5444	0.0704
His			0.3354	
Ile	0.0383	1.039^{***}	0.4943	
Leu			0.4148	
Lys	−0.0586	0.9431^{***}	0.318	0.0674
Phe			0.3677	
Pro	−0.0179	0.9823^{*}	0.3945	0.0546
Ser	0.0427	1.0436^{***}	0.3343	0.0827
Thr	0.0437	1.0447^{***}	0.3476	0.0814
Tyr	0.0323	1.0328^{***}	0.3456	
Val	0.0545	1.056^{***}	0.4056	0.1109

Note —AA: amino acid; random effects are shown as "standard deviations"
Significance levels: $^{***}P < 0.001$; $^{**}P < 0.01$;$^{*}P < 0.05$.
N.S. nonsignificant, but in the model because its random slope is in the model.

Table 4.9: Results for *D.melanogaster (fly)* by considering only *agg*

AA	*Fixed effects*		*Random Effects (standard deviation)*	
	coefficients	Odds Ratio	Random Intercept	Random Slope
Ala	0.1875	1.2062^{***}	0.412	0.0978
Arg	0.1266	1.135^{***}	0.4757	0.0789
Asn	0.1135	1.1202^{***}	0.4605	
Asp	0.117	1.1241^{***}	0.4098	
Cys	0.107	1.1129^{***}	0.421	
Gln	-0.0074	$0.9926^{N.S.}$	0.509	0.0693
Glu	0.0215	1.0217^{**}	0.5656	0.0979
Gly	0.1144	1.1212^{***}	0.3921	0.0614
His	0.1626	1.1766^{***}	0.377	
Ile	0.0751	1.078^{***}	0.5348	0.0372
Leu	0.0233	1.0236^{***}	0.598	
Lys	-0.0136	$0.9865^{N.S.}$	0.6122	0.0876
Phe	0.0448	1.0458^{***}	0.5585	
Pro	0.2287	1.257^{***}	0.3402	
Ser	0.0978	1.1027^{***}	0.4549	0.0376
Thr	0.1881	1.207^{***}	0.4202	0.0631
Tyr	0.1198	1.1273^{***}	0.4507	
Val	0.039	1.0398^{***}	0.4835	0.0758

Note —AA: amino acid; random effects are shown as "standard deviations"
Significance levels: $^{***}P < 0.001$; $^{**}P < 0.01$;$^{*}P < 0.05$.
$N.S.$ nonsignificant, but in the model because its random slope is in the model.

Table 4.10: Results for *E. coli* by considering only *acc* and *ent*

	Fixed Effects				*Random Effects(standard deviation)*	
	acc		*ent*		*Random Intercept*	*Random Slope(s)*
AA	coefficients	Odds Ratio	coefficients	Odds Ratio		
Ala	0.055	$1.0565^{N.S.}$	0.0354	$1.036^{N.S.}$	0.5197	ent : 0.2333; acc : 0.4327
Arg	0.1154	$1.1223^{N.S.}$	-0.6456	0.5243^{***}	0.8837	ent : 0.2957; acc : 0.3230
Asn	-0.4951	0.6095^{***}	-0.5041	0.604^{***}	0.8057	ent : 0.2039; acc : 0.0964
Asp	-0.0231	$0.9772^{N.S.}$	-0.0261	$0.9742^{N.S.}$	0.4724	ent : 0.1141; acc : 0.1815
Cys					0.3103	
Gln			-0.2234	0.7998^{***}	0.3523	
Glu			-0.0545	0.947^{*}	0.2726	
Gly			-0.171	0.8428^{***}	0.6449	
His	-0.0289	$0.9715^{N.S.}$	-0.2978	0.7424^{***}	0.674	ent : 0.1355; acc : 0.4204
Ile	0.0592	$1.061^{N.S.}$	-0.1339	0.8747^{***}	0.5639	ent : 0.1375; acc : 0.0420
Leu	-0.2081	0.8121^{**}	-0.1831	0.8327^{***}	0.5668	
Lys*						
Phe			-0.1391	0.8701^{***}	0.5905	
Pro			-0.2237	0.7996^{***}	0.4674	
Ser	-0.1431	$0.8667^{N.S.}$	-0.3937	0.6746^{***}	0.7144	ent : 0.2162; acc : 0.3253
Thr	-0.1108	$0.8951^{N.S.}$	-0.2221	0.8008^{***}	0.5489	ent : 0.1542; acc : 0.2297
Tyr					0.4556	
Val	0.2803	1.3235^{***}	-0.0359	$0.9647^{N.S.}$	0.5676	ent : 0.1429; acc : 0.2691

Note —AA: amino acid; random effects are shown as "standard deviations"
Significance levels: $^{***}P < 0.001$; $^{**}P < 0.01$;$^{*}P < 0.05$.
$N.S.$ nonsignificant, but in the model because its random slope is in the model.
Lys*: no optimal codon.

Table 4.11: Results for *S.cerevisiae (yeast)* by considering only *acc* and *ent*

	Fixed Effects				*Random Effects(standard deviation)*	
	acc		*ent*		*Random Intercept*	*Random Slope(s)*
AA	coefficients	Odds Ratio	coefficients	Odds Ratio		
Ala	-0.0443	$0.9567^{N.S.}$	-0.1947	0.8231^{***}	0.604	$ent: 0.2081; acc: 0.2012$
Arg	0.1098	$1.1161^{N.S.}$	-0.4523	0.6362^{***}	0.6513	$ent: 0.3281; acc: 0.0868$
Asn	0.1536	1.166^{*}	-0.0953	0.9091^{***}	0.5931	$ent: 0.1274; acc: 0.2876$
Asp					0.3247	
Cys			-0.2417	0.7853^{***}	0.3117	
Gln			-0.0587	0.943^{*}	0.4472	
Glu			-0.1	0.9048^{***}	0.3169	
Gly	-0.2547	0.7751^{***}	-0.3651	0.6941^{***}	0.7378	$ent: 0.3516; acc: 0.2774$
His					0.3354	
Ile	-0.1165	$0.89^{N.S.}$	-0.3772	0.6858^{***}	0.8109	$ent: 0.2251; acc: 0.3449$
Leu	0.0448	$1.0458^{N.S.}$	-0.1704	0.8433^{***}	0.448	$ent: 0.162; acc: 0.2193$
Lys	0.0749	$1.0778^{N.S.}$	-0.1108	0.8951^{***}	0.4829	$ent: 0.1206; acc: 0.0252$
Phe	-0.0207	$0.9795^{N.S.}$	-0.0724	$0.9302^{N.S.}$	0.4768	$ent: 0.2848; acc: 0.5157$
Pro	0.1447	$1.1557^{N.S.}$	-0.0729	$0.9297^{N.S.}$	0.4801	$ent: 0.161; acc: 0.2457$
Ser	0.0384	$1.0391^{N.S.}$	-0.3986	0.6713^{***}	0.5132	$ent: 0.2303; acc: 0.1045$
Thr	0.0106	$1.0107^{N.S.}$	-0.0749	0.9278^{*}	0.535	$ent: 0.2158; acc: 0.2494$
Tyr			-0.1192	0.8876^{***}	0.3722	
Val	-0.1215	$0.8856^{N.S.}$	-0.146	0.8642^{***}	0.6475	$ent: 0.2421; acc: 0.1191$

Note —AA: amino acid; random effects are shown as "standard deviations"
Significance levels: $^{***}P < 0.001$; $^{**}P < 0.01$;$^{*}P < 0.05$.
$N.S.$ nonsignificant, but in the model because its random slope is in the model.

Table 4.12: Results for *D.melanogaster (fly)* by considering only *acc* and *ent*

	Fixed Effects				*Random Effects(standard deviation)*	
	acc		*ent*		*Random Intercept*	*Random Slope(s)*
AA	coefficients	Odds Ratio	coefficients	Odds Ratio		
Ala	−0.0803	$0.9228^{N.S.}$	−0.2621	0.7694^{***}	0.3616	$ent : 0.0087; acc : 0.1028$
Arg	0.0764	$1.0794^{N.S.}$	−0.4445	0.7694^{***}	0.4247	$ent : 0.2482; acc : 0.1574$
Asn			−0.0889	0.6411^{***}	0.3985	
Asp	0.0663	$1.0685^{N.S.}$	−0.1505	0.9149^{***}	0.4238	$ent : 0.1083; acc : 0.0745$
Cys			−0.1583	0.8603^{***}	0.3155	$acc :$
Gln	0.0736	$1.0764^{N.S.}$	−0.167	0.8464^{***}	0.4362	$ent : 0.2107; acc : 0.2654$
Glu			−0.2804	0.8462^{***}	0.4138	
Gly	0.0852	1.0889^{*}	−0.1156	0.7555^{***}	0.3597	
His			−0.1185	0.8908^{***}	0.319	
Ile			−0.1545	0.8883^{***}	0.4153	
Leu			−0.2557	0.8568^{***}	0.4658	
Lys			−0.2484	0.7800^{***}	0.4797	
Phe			−0.1634	0.8492^{***}	0.4897	
Pro			−0.228	0.7961^{***}	0.3089	
Ser	−0.0273	$0.9731^{N.S.}$	−0.1668	0.7961^{***}	0.4787	$ent : 0.1369; acc : 0.1289$
Thr	−0.0696	$0.9328^{N.S.}$	−0.1001	0.8464^{***}	0.4191	$ent : 0.1533; acc : 0.1242$
Tyr			−0.1311	0.9047^{***}	0.3843	
Val			−0.226	0.8771^{***}	0.3829	

Note —AA: amino acid; random effects are shown as "standard deviations"
Significance levels: $^{***}P < 0.001$; $^{**}P < 0.01$;$^{*}P < 0.05$.
$N.S.$ nonsignificant, but in the model because its random slope is in the model.

Table 4.13: Results for *E. coli* by considering only *acc* and *agg*

	Fixed Effects				*Random Effects(standard deviation)*	
	acc		*agg*		*Random Intercept*	*Random Slope(s)*
AA	coefficients	Odds Ratio	coefficients	Odds Ratio		
Ala	0.1855	1.2038^{***}	0.0395	1.0403^{*}	0.4707	
Arg	−0.1466	0.8636^{*}	−0.0206	$0.9796^{N.S.}$	0.9983	$acc : 0.3986; agg : 0.0708$
Asn	−0.279	0.7565^{***}	0.0674	1.0697^{***}	0.7969	$acc : 0.3173; agg : 0.0299$
Asp	0.0751	$1.078^{N.S.}$	0.0788	1.082^{***}	0.4348	$acc : 0.1648; agg : 0.0687$
Cys			−0.0433	0.9576^{*}	0.3184	
Gln	−0.1908	0.8263^{***}	0.0435	1.0445^{**}	0.4333	$acc : 0.4928; agg : 0.0684$
Glu	−0.086	$0.9176^{N.S.}$	0.069	1.0714^{**}	0.3364	
Gly	−0.0644	$0.9376^{N.S.}$	0.1377	1.1476^{***}	0.6106	
His	0.0641	$1.0662^{N.S.}$	0.119	1.1264^{***}	0.6077	$acc : 0.3389; agg : 0.0234$
Ile			0.0356	1.0362^{***}	0.51	
Leu	−0.2429	0.7843^{***}	−0.0452	0.9558^{***}	0.6357	$acc : 0.0498; agg : 0.0444$
Lys*						
Phe					0.5582	
Pro	−0.2518	0.7774^{***}			0.5948	
Ser	−0.314	0.7305^{***}	0.0619	1.0639^{***}	0.554	
Thr	−0.2097	0.8108^{***}	0.1685	1.1835^{***}	0.5203	
Tyr			0.0944	1.099^{***}	0.4676	
Val	0.1935	1.2135^{***}	−0.094	0.9103^{***}	0.4805	

Note —AA: amino acid; random effects are shown as "standard deviations"
Significance levels: $^{***}P < 0.001$; $^{**}P < 0.01$;$^{*}P < 0.05$.
N.S. nonsignificant, but in the model because its random slope is in the model.
Lys*: no optimal codon.

Table 4.14: Results for *S.cerevisiae (yeast)* by considering only *acc* and *agg*

	Fixed Effects				*Random Effects(standard deviation)*	
	acc		*agg*		*Random Intercept*	*Random Slope(s)*
AA	coefficients	Odds Ratio	coefficients	Odds Ratio		
Ala			0.0264	1.0268^{***}	0.4171	
Arg	-0.1005	$0.9044^{N.S.}$	0.034	1.0346^{**}	0.5659	$acc : 0.2364; agg : 0.02$
Asn			0.0274	1.0278^{***}	0.3187	
Asp	0.0949	1.0995^{***}			0.367	
Cys	-0.1999	0.8188^{**}			0.4042	
Gln					0.3545	
Glu			0.0175	1.0177^{**}	0.246	
Gly	-0.1912	0.826^{***}	0.0656	1.0678^{***}	0.693	$acc : 0.2589; agg : 0.0684$
His					0.3354	
Ile	-0.2049	0.8147^{***}	0.0111	$1.0112^{N.S.}$	0.612	$acc : 0.2666; agg : 0.1058$
Leu	-0.0936	0.9106^{**}			0.4657	
Lys			-0.0596	0.9421^{***}	0.3147	
Phe					0.3677	
Pro	-0.0911	$0.9129^{N.S.}$	-0.0197	$0.9805^{N.S.}$	0.4893	$acc : 0.0857; agg : 0.0736$
Ser	-0.1717	0.8422^{***}	0.0424	1.0433^{***}	0.4076	
Thr			0.0436	1.0446^{***}	0.3531	
Tyr			0.0323	1.0328^{***}	0.3456	
Val	-0.1147	0.8916^{*}	0.0503	1.0516^{***}	0.4694	$acc : 0.0956; agg : 0.1033$

Note —AA: amino acid; random effects are shown as "standard deviations"
Significance levels: $^{***}P < 0.001$; $^{**}P < 0.01$;$^{*}P < 0.05$.
$N.S.$ nonsignificant, but in the model because its random slope is in the model.

Table 4.15: Results for *D.melanogaster(fly)* by considering only *acc* and *agg*

	Fixed Effects				*Random Effects(standard deviation)*	
	acc		*agg*		*Random Intercept*	*Random Slope(s)*
AA	coefficients	Odds Ratio	coefficients	Odds Ratio		
Ala	-0.0413	$0.9595^{N.S.}$	0.2104	1.2342^{***}	0.4446	acc : 0.2851; $Ragg$: 0.1128
Arg	0.0186	$1.0188^{N.S.}$	0.1258	1.1341^{***}	0.4003	acc : 0.1822; $Ragg$: 0.053
Asn			0.1135	1.1202^{***}	0.4605	
Asp	0.145	1.156^{**}	0.1563	1.1692^{***}	0.4281	
Cys			0.107	1.1129^{***}	0.421	
Gln					0.5256	
Glu			0.0239	1.0242^{***}	0.5616	
Gly	0.1033	1.1088^{*}	0.1249	1.133^{***}	0.4329	acc : 0.0856; $Ragg$: 0.0681
His			0.1626	1.1766^{***}	0.377	
Ile	0.0068	$1.0068^{N.S.}$	0.0692	1.0717^{***}	0.5773	acc : 0.0574; $Ragg$: 0.0402
Leu			0.0233	1.0236^{***}	0.598	
Lys					0.6007	
Phe			0.0448	1.0458^{***}	0.5585	
Pro	0.1195	1.1269^{*}	0.2127	1.237^{***}	0.3796	
Ser			0.0929	1.0974^{***}	0.4541	
Thr			0.1862	1.2047^{***}	0.4266	
Tyr			0.1198	1.1273^{***}	0.4507	
Val	0.0419	$1.0428^{N.S.}$	0.0364	1.0371^{**}	0.4796	acc : 0.0837; $Ragg$: 0.1227

Note —AA: amino acid; random effects are shown as "standard deviations"
Significance levels: $^{***}P < 0.001$; $^{**}P < 0.01$;$^{*}P < 0.05$.
$N.S.$ nonsignificant, but in the model because its random slope is in the model.

Table 4.16: Results for *E. coli* by considering only *ent* and *agg*

	Fixed Effects				*Random Effects(standard deviation)*	
	agg		*ent*		*Random Intercept*	*Random Slope(s)*
AA	coefficients	Odds Ratio	coefficients	Odds Ratio		
Ala	0.0119	$1.012^{N.S.}$	0.0545	$1.056^{N.S.}$	0.5744	$ent: 0.1759; agg: 0.0837$
Arg	−0.1311	0.8771***	−0.6325	1.056***	0.8823	$ent: 0.3756; agg: 0.1812$
Asn	0.0853	1.089***	−0.3773	0.5313***	0.7246	$ent: 0.1552; agg: 0.0307$
Asp	0.04	1.0408*	-0.045	$0.6857^{N.S.}$	0.4481	$ent: 0.1097; agg: 0.1$
Cys	−0.0433	0.9576*			0.3184	
Gln			−0.2234	0.7998***	0.3523	
Glu	0.0295	$1.0299^{N.S.}$	-0.0369	$0.7998^{N.S.}$	0.3032	$ent: 0.1453; agg: 0.0964$
Gly	0.1422	1.1528***	−0.1136	0.9638**	0.676	
His	0.1186	1.1259***	−0.2686	0.8926***	0.5243	
Ile	0.0517	1.0531**	−0.1134	0.7644***	0.5842	$ent: 0.0994; agg: 0.0525$
Leu	−0.041	0.9598**	−0.2223	0.8928***	0.5803	$ent: 0.1314; agg: 0.0505$
Lys*						
Phe			−0.1391	0.8701***	0.5905	
Pro			−0.2237	0.7996***	0.4674	
Ser**	0.1552	1.1679***	−0.3429	0.7996***	0.7233	$ent: 0.2554; agg: 0.0589$
Thr	0.137	1.1468***	−0.2348	0.7097***	0.6385	$ent: 0.1874; agg: 0.0756$
Tyr	0.0768	1.0789**	-0.0893	$0.7907^{N.S.}$	0.3745	$ent: 0.2018; agg: 0.0748$
Val	−0.0899	0.914***	-0.0147	$0.9146^{N.S.}$	0.5914	$ent: 0.1935; agg: 0.0718$

Note —AA: amino acid; random effects are shown as "standard deviations"
Significance levels: $^{***}P < 0.001$; $^{**}P < 0.01$;$^{*}P < 0.05$.
N.S. nonsignificant, but in the model because its random slope is in the model.
Lys*: no optimal codon.
Ser**: agg:ent interaction is significant and estimate coefficient is -0.123.

Table 4.17: Results for *S.cerevisiae (yeast)* by considering only *ent* and agg

	Fixed Effects				*Random Effects(standard deviation)*	
	agg		*ent*		*Random Intercept*	*Random Slope(s)*
AA	coefficients	Odds Ratio	coefficients	Odds Ratio		
Ala	0.018	$1.0182^{N.S.}$	-0.1149	0.8915^{***}	0.5121	$ent: 0.1824; agg: 0.0647$
Arg	-0.0057	$0.9943^{N.S.}$	-0.3529	0.7026^{***}	0.6091	$ent: 0.2834; agg: 0.0812$
Asn			-0.0772	0.9257^{***}	0.3771	$ent :; agg :$
Asp					0.3247	$ent :; agg :$
Cys	-0.0324	$0.9681^{N.S.}$	-0.1812	0.8343^{**}	0.43	$ent: 0.2624; agg: 0.0112$
Gln					0.3545	$ent :; agg :$
Glu	-0.0216	$0.9786^{N.S.}$	-0.1022	0.9028^{***}	0.3459	$ent: 0.2054; agg: 0.0222$
Gly	0.0163	$1.0164^{N.S.}$	-0.3224	0.7244^{***}	0.5982	$ent: 0.3385; agg: 0.0206$
His					0.3354	$ent :; agg :$
Ile	-0.0155	$0.9846^{N.S.}$	-0.3286	0.7199^{***}	0.6449	$ent: 0.2171; agg: 0.074$
Leu	-0.0104	$0.9897^{N.S.}$	-0.1266	0.8811^{***}	0.3461	$ent: 0.096; agg: 0.0252$
Lys	-0.0637	0.9383^{***}	-0.0921	0.912^{***}	0.421	$ent: 0.0974; agg: 0.0661$
Phe			-0.0781	0.9249^{**}	0.3546	$ent :; agg :$
Pro			-0.0889	0.9149^{***}	0.4276	$ent :; agg :$
Ser	0.0068	$1.0068^{N.S.}$	-0.3624	0.696^{***}	0.4292	$ent: 0.1915; agg: 0.0761$
Thr	0.042	1.0429^{*}	-0.0377	$0.963^{N.S.}$	0.4357	$ent: 0.0973; agg: 0.0365$
Tyr			-0.1192	0.8876^{***}	0.3722	$ent :; agg :$
Val	0.0284	$1.0288^{N.S.}$	-0.1289	0.8791^{***}	0.546	$ent: 0.1756; agg: 0.1358$

Note —AA: amino acid; random effects are shown as "standard deviations"
Significance levels: $^{***}P < 0.001$; $^{**}P < 0.01$; $^{*}P < 0.05$.
$N.S.$ nonsignificant, but in the model because its random slope is in the model.

Table 4.18: Results for *D.melanogaster(fly)* by considering only *ent* and *agg*

	Fixed Effects				*Random Effects(standard deviation)*	
	agg		*ent*		*Random Intercept*	*Random Slope(s)*
AA	coefficients	Odds Ratio	coefficients	Odds Ratio		
Ala	0.1756	1.192^{***}	-0.2154	0.8062^{***}	0.364	$ent: 0.1711; agg: 0.0237$
Arg	0.1317	1.1408^{***}	-0.5075	0.602^{***}	0.3826	$ent: 0.1865; agg: 0.1152$
Asn	0.1093	1.1155^{***}	-0.0901	0.9138^{**}	0.4175	
Asp	0.1112	1.1176^{***}	-0.1732	0.841^{***}	0.3712	
Cys	0.107	1.1129^{***}			0.421	
Gln	-0.0004	0.9996^{***}	-0.1438	0.8661^{***}	0.3462	$ent: 0.1557; agg: 0.09$
Glu	0.0033	1.0033^{***}	-0.3107	0.7329^{***}	0.4375	$ent: 0.1191; agg: 0.1227$
Gly	0.1073	1.1133^{***}	-0.15	0.8607^{***}	0.3343	$ent: 0.0978; agg: 0.0874$
His	0.1594	1.1728^{***}	-0.1036	0.9016^{*}	0.3236	
Ile	0.0572	1.0589^{***}	-0.1738	0.8405^{***}	0.4591	
Leu			-0.2557	0.7744^{***}	0.4658	
Lys	-0.036	$0.9646^{N.S.}$	-0.3223	0.7245^{***}	0.5294	$ent: 0.2105; agg: 0.0999$
Phe	0.0128	$1.0129^{N.S.}$	-0.2323	0.7927^{***}	0.5223	$ent: 0.2038; agg: 0.0808$
Pro	0.1995	1.2208^{***}	-0.2257	0.798^{***}	0.3271	
Ser	0.076	1.079^{***}	-0.1699	0.8437^{***}	0.3925	$ent: 0.1348; agg: 0.0772$
Thr	0.1671	1.1819^{***}	-0.0939	0.9104^{***}	0.4425	$ent: 0.0919; agg: 0.0173$
Tyr	0.1198	1.1273^{***}			0.4507	
Val	0.0238	$1.0241^{N.S.}$	-0.1915	0.8257^{***}	0.4283	$ent: 0.1784; agg: 0.0733$

Note —AA: amino acid; random effects are shown as "standard deviations"
Significance levels: $^{***}P < 0.001$; $^{**}P < 0.01$; $^{*}P < 0.05$.
N.S. nonsignificant, but in the model because its random slope is in the model.

Table 4.19: Results for *E. coli* considering all three factors

	Fixed Effects						*Random Effects(standard deviation)*	
	acc		*agg*		*ent*		*Random Intercept*	*Random Slope(s)*
AA	coefficients	Odds Ratio	coefficients	Odds Ratio	coefficients	Odds Ratio		
Ala					0.0608	1.0627 *	0.546	*ent* : 0.1656
Arg			−0.1311	0.8771 ***	−0.6325	0.5313 ***	0.8823	*ent* : 0.3756; *agg* : 0.1812
Asn	−0.4038	0.6678 **	0.0861	1.0899 ***	−0.5262	0.5908 ***	0.7849	*ent* : 0.2003
Asp			0.0362	1.0369 ***			0.4038	*agg* : 0.0831
Cys							0.3103	
Gln					−0.2234	0.7998 ***	0.3523	
Glu					−0.0545	0.947 *	0.2726	
Gly			0.1165	1.1236 ***	−0.1202	0.8867 **	0.675	
His			0.1186	1.1259 ***	−0.2686	0.7644 ***	0.5243	
Ile			0.0492	1.0504 **	−0.1148	0.8915 ***	0.5718	*ent* : 0.1004
Leu	−0.1872	0.8293 *	−0.0502	0.951 ***	−0.1868	0.8296 ***	0.5796	*agg* : 0.0536
Lys*								
Phe					−0.1391	0.8701 ***	0.5905	
Pro					−0.2237	0.7996 ***	0.4674	
Ser**			0.1549	1.1675 ***	−0.3428	0.7098 ***	0.7232	*ent* : 0.2541; *agg* : 0.0574
Thr			0.1408	1.1512 ***	−0.2337	0.7916 ***	0.6277	*ent* : 0.179
Tyr			0.0944	1.099 ***			0.4676	
Val	0.2981	1.3473 ***	−0.0905	0.9135 ***	−0.0465	0.9546 *N.S.*	0.5709	*ent* : 0.1948

Note —AA: amino acid; random effects are shown as "standard deviations"
Significance levels: $^{***}P < 0.001$; $^{**}P < 0.01$;$^{*}P < 0.05$.
N.S. nonsignificant, but in the model because its random slope is in the model.
Lys*: no optimal codon.
Ser**: agg:ent interaction is significant and estimate coefficient is -0.123.

Table 4.20: Results for *S.cerevisiae (yeast)* considering all three factors

	Fixed Effects						*Random Effects(standard deviation)*	
	acc		*agg*		*ent*			
AA	coefficients	Odds Ratio	coefficients	Odds Ratio	coefficients	Odds Ratio	*Random Intercept*	*Random Slope(s)*
Ala					−0.1835	0.8324 ***	0.6329	*ent* : 0.1672
Arg					−0.3929	0.6751 ***	0.641	*ent* : 0.3334
Asn					−0.0772	0.9257 ***	0.3771	
Asp							0.3247	
Cys					−0.2417	0.7853 ***	0.3117	
Gln					−0.0587	0.943 *	0.4472	
Glu					−0.1	0.9048 ***	0.3169	
Gly	−0.2629	0.7688 ***			−0.3657	0.6937 ***	0.7091	*ent* : 0.3659
His							0.3354	
Ile			−0.0155	0.9846 $^{N.S.}$	−0.3286	0.7199 ***	0.6449	*ent* : 0.2171; *agg* : 0.074
Leu					−0.1576	0.8542 ***	0.3791	
Lys			−0.0645	0.9375 ***	−0.0856	0.918 ***	0.363	
Phe					−0.0825	0.9208 **	0.3824	*ent* : 0.2092
Pro					−0.0889	0.9149 ***	0.4276	
Ser	0.0383	1.039 $^{N.S.}$			−0.3986	0.6713 ***	0.5132	*ent* : 0.2303; *acc* : 0.1045
Thr			0.0421	1.043 *	−0.0377	0.963 $^{N.S.}$	0.4328	*ent* : 0.0964
Tyr					−0.1192	0.8876 ***	0.3722	
Val					−0.1663	0.8468 ***	0.6005	*ent* : 0.2071

Note —AA: amino acid; random effects are shown as "standard deviations"
Significance levels: $^{***}P < 0.001$; $^{**}P < 0.01$; $^{*}P < 0.05$.
N.S. nonsignificant, but in the model because its random slope is in the model.

Table 4.21: Results for *D.melanogaster (fly)* considering all three factors

	Fixed Effects						*Random Effects(standard deviation)*	
	acc		*agg*		*ent*			
AA	coefficients	Odds Ratio	coefficients	Odds Ratio	coefficients	Odds Ratio	*Random Intercept*	*Random Slope(s)*
Ala			0.1778	1.1946 ***	−0.2183	0.8039 ***	0.3453	
Arg	0.2264	1.2541 *	0.1459	1.1571 ***	−0.5407	0.5823 ***	0.4101	
Asn			0.1093	1.1155 ***	−0.0901	0.9138 **	0.4175	
Asp			0.1112	1.1176 ***	−0.1732	0.841 ***	0.3712	
Cys			0.107	1.1129 ***			0.421	
Gln					−0.1648	0.8481 ***	0.4081	
Glu					−0.2804	0.7555 ***	0.4138	
Gly	0.2146	1.2394 **	0.127	1.1354 ***	−0.0823	0.921 $^{N.S.}$	0.3596	*ent* : 0.202
His			0.1594	1.1728 ***	−0.1036	0.9016 *	0.3236	
Ile			0.0572	1.0589 ***	−0.1738	0.8405 ***	0.4591	
Leu					−0.2557	0.7744 ***	0.4658	
Lys					−0.2484	0.78 ***	0.4797	
Phe					−0.1634	0.8493 ***	0.4897	
Pro	0.256	1.2918 *	0.1971	1.2179 ***	−0.1949	0.8229 ***	0.3664	
Ser			0.0825	1.086 ***	−0.1679	0.8454 ***	0.3918	*ent* : 0.1257
Thr			0.1681	1.1831 ***	−0.0815	0.9217 **	0.4026	
Tyr			0.1198	1.1273 ***			0.4507	
Val			0.0238	1.0241 $^{N.S.}$	−0.1751	0.8394 ***	0.3885	*agg* : 0.0787

Note —AA: amino acid; random effects are shown as "standard deviations"
Significance levels: $^{***}P < 0.001$; $^{**}P < 0.01$; $^{*}P < 0.05$.
$N.S.$ nonsignificant, but in the model because its random slope is in the model.

Appendix

R code

```
require("lme4")
require("multcomp")
species <- "fly"
#species <- "yeast"
#species <- "ecoli"
read.data <- T
#read.data <- F # comment out to not read in data again

prepareData <- function( d )
{
# replace hyphens with NAs. They are the same for this analysis
# sovlent accessibility
x <- d$acc=='-'
d$acc[x] <- NA
d$acc <- as.numeric( as.character( d$acc ) )

# functional sites
x <- d$func=='-'
d$func[x] <- NA
d$func <- factor( d$func )

# create a dichotomous variable for optimal/non-optimal codons.
# it is defined by testing if a optimality is significantly bigger
# than 1, here a cut-value is used to define optimal/non-optimal
# codons according to the testing results
  # for E.coli: 1.189 or larger YES; 1.136or smaller NO,  choose 1.14
  # as the cut-value,for yeast, 1.261 is the smallest optimal value for
  # being optimal. so 1 can be the cut-value,
# so is 1.14. 1.14 can be the cut-value for fly, too. (1.135or smaller, NO; 1.198 or larger  Yes),
  # except for amino acid Ser: TCC : 1.113, which is smaller than 1.135 but YES as optimal.
  # so Ser is claasified seperately.
cbind( d, optcodon = factor( d$opt>1.14 ) )
}

if (read.data)
{
cat( "reading data for", species, "\n" )
all <- read.table( paste( "../report/data/", species,
"/", species, "-all.dat.gz", sep='' ), header=T )

aaA <- all[all$aa=='A',]
aaC <- all[all$aa=='C',]
aaD <- all[all$aa=='D',]
aaE <- all[all$aa=='E',]
aaF <- all[all$aa=='F',]
aaG <- all[all$aa=='G',]
aaH <- all[all$aa=='H',]
aaI <- all[all$aa=='I',]
aaK <- all[all$aa=='K',]
aaL <- all[all$aa=='L',]
aaM <- all[all$aa=='M',]
aaN <- all[all$aa=='N',]
aaP <- all[all$aa=='P',]
```

```
aaQ <- all[all$aa=='Q',]
aaR <- all[all$aa=='R',]
aaS <- all[all$aa=='S',]
aaT <- all[all$aa=='T',]
aaV <- all[all$aa=='V',]
aaW <- all[all$aa=='W',]
  aaY <- all[all$aa=='Y',]

aaA <- prepareData( aaA )
aaC <- prepareData( aaC )
aaD <- prepareData( aaD )
aaE <- prepareData( aaE )
aaF <- prepareData( aaF )
aaG <- prepareData( aaG )
aaH <- prepareData( aaH )
aaI <- prepareData( aaI )
aaK <- prepareData( aaK )
aaL <- prepareData( aaL )
aaM <- prepareData( aaM )
aaN <- prepareData( aaN )
aaP <- prepareData( aaP )
aaQ <- prepareData( aaQ )
aaR <- prepareData( aaR )
aaS <- prepareData( aaS )
aaT <- prepareData( aaT )
aaV <- prepareData( aaV )
aaW <- prepareData( aaW )
aaY <- prepareData( aaY )
cat( "all data read\n" )
  if(species == 'fly')
  {aaS$optcodon = factor( aaS$opt>1.1)}
}


random.effect.selection <- function(data, response='optcodon',
fixedstr = 'acc*ent*agg',rslopeterms= c('acc', 'agg', 'ent'))

# data: the data frame to analyze
# response: the response variable, e.g. 'optcodon'
# fixedstr : fixed effect structure eg: fixedstr = 'acc*ent*agg'
# rslopeterms: possible random slope terms  eg: 'acc', 'agg', 'ent'

{
keep.always=c()
  linear= c()
  randomslopeterms = c()
  expr = parse( text = paste( "glmer(",response,"~",fixedstr,
  "+ (1|orf), data=data,family=binomial)", sep='' ) )

m = eval(expr)
aic0 = AIC(logLik(m))
  cat("1|orf  AIC :",aic0,"\n")

  rslope1 = ''
for (p in rslopeterms) {
```

```
expr = parse( text = paste( "glmer(",response,"~",
fixedstr,"+ (", p, "|orf),data=data, family=binomial)", sep='' ) )

m = eval(expr)
aic = AIC(logLik(m))
  cat(p,"|orf  AIC :",aic,"\n")
  if (aic < aic0) {
rslope1 = p
aic0 = aic
}
}
if (rslope1 == '')
{
    keep.always=c()
    linear= rslopeterms
    randomslopeterms = c()
    random = '(1|orf)'
cat("keep.always terms are : ", keep.always, "\n")
    cat("linear terms are : ", linear, "\n")
    cat("random terms are : ", random, "\n")
    return( list( keep=keep.always,linear =linear,random =  random,
    randomslopeterms= randomslopeterms  ) )
}

else {
rslope2 = ''
for (p in rslopeterms)
{
if (p != rslope1) {
expr = parse( text = paste( "glmer(optcodon ~ acc*ent*agg +
(", p, "+", rslope1, "|orf), data=data, family=binomial)", sep='' ) )
m = eval(expr)
aic = AIC(logLik(m))
    cat(p,"+", rslope1,"|orf  AIC :",aic,"\n")

if (aic < aic0)
{
rslope2 = p
aic0 = aic
}
}
}

if (rslope2 == '') {
    randomslopeterms = rslope1
    random = paste("(", rslope1,"|orf)",  sep='')
keep.always= rslope1
    for (p in rslopeterms) {
if ( p != rslope1) {
  linear= c(linear,p)
}
}

     cat("keep.always terms are : ", keep.always, "\n")
       cat("linear terms are : ", linear, "\n")
       cat("random terms are : ", random, "\n")
```

```
       return( list( keep=keep.always,linear =linear,random = random,
       randomslopeterms= randomslopeterms  ) )

}
else {
rslope3 = ''
for (p in rslopeterms) {
if (p != rslope1 && p != rslope2) {
expr = parse( text = paste( "glmer(optcodon ~ acc*ent*agg + (", p, "+", rslope2,
"+", rslope1, "|orf), data=data, family=binomial)", sep='' ) )

m = eval(expr)
aic = AIC(logLik(m))
  cat(p,"+", rslope1,"+", rslope2,"|orf  AIC :",aic,"\n")

if (aic < aic0) {
rslope3 = p
aic0 = aic
}
}
}

if (rslope3 == '') {
  randomslopeterms = c(rslope1,rslope2)
  random = paste("(", rslope1,"+",rslope2, "|orf)" , sep='')
keep.always= c(keep.always, rslope1,rslope2)
  for (p in rslopeterms) {
if ( p != rslope1 && p!= rslope2 ) {
linear= c(linear,p)

}
}
  cat("keep.always terms are : ", keep.always, "\n")
  cat("linear terms are : ", linear, "\n")
  cat("random terms are : ", random, "\n")
  return( list( keep=keep.always,linear =linear,random =  random,
  randomslopeterms= randomslopeterms  ) )
}

else {
   randomslopeterms = c(rslope1,rslope2,rslope3)
   random = paste("(",rslope1,"+",rslope2,"+", rslope3, "|orf)" , sep='')
 keep.always = rslopeterms
   linear = c()
   cat("keep.always terms are : ", keep.always, "\n")
   cat("linear terms are : ", linear, "\n")
   cat("random terms are : ", random, "\n")
   return( list( keep=keep.always,linear =linear,random =  random,
   randomslopeterms= randomslopeterms  ) )

}
}
}

}
```

```
fixed.model.selection <- function( data, response, keep.always=c(),
 linear, quadratic=c(), cubic=c(), random='(1|orf)', p.cutoff=.05 )
# data: the data frame to analyze
# response: the response variable, e.g. 'optcodon'
# keep.always: any terms that should never be removed from the model,
# e.g. because they exist in the random structure.
# For now, this works only for linear terms! Also, a term that is listed
# in keep.always must not show up in linear.
# linear: the linear terms
# quadratic: the quadratic terms. Note that the function assumes that
# all predictors that appear in quadratic terms also appear in linear terms.
# If you don't satisfy this assumption, thinks can (and will) go wrong.
# cubic: the cubic terms. The same caveat applies as to quadratic terms.
# All variables have to appear in all lower-order combinations.
# random: the random-effect structure
# p.cutoff: the cutoff below which terms are kept

{
# first we build a string of the model we want to analyze
terms <- c( 'Intercept', keep.always, linear, quadratic, cubic ) # terms in our model
model.formula <- paste( response, "~", paste( terms[-1], collapse=' + '),"+", random )

if ( length( terms ) == 1 ){
cat("\nCannot evaluate mixed linear model without any fixed effects ->
no significant terms survive\n")

   expr = parse( text = paste( "glmer(optcodon ~ 1 + ",random,", data=data, family=binomial)", sep='' ) )
   m0 <- eval(expr)
   return( list( formula=model.formula, model=m0, multcomp.summary=list()  ) )
}

cat( "\nAnalyzing model:", model.formula, "\n" )
# now build the entire glmer expression and evaluate it
expr <- parse( text = paste( "glmer(", model.formula, ", data=data, family=binomial)", sep='' ) )
m <- eval(expr)
#print(summary(m))
# build the linear-function comparison matrix
K <- diag(length(terms))
rownames(K) <- terms
# calculate p values
lhs <- summary( glht( m, linfct = K ) )
print( lhs )
if ( length( cubic ) > 0 ) # test for cubic terms
{
# extract p values corresponding to cubic terms
p <- lhs$test$pvalues[(length(terms)-length(cubic)+1):length(terms)]
i <- which.max(p) # find the term with the smallest p values
if ( p[i] >= p.cutoff ) # is it larger than the cutoff?
{
# yes, remove term
cat("\nRemoving term", cubic[i], "with p value", p[i], ">=", p.cutoff, "\n" )
cubic <- cubic[-i]
# now, recursively remove other terms
fixed.model.selection( data, response, keep.always, linear, quadratic, cubic, random, p.cutoff )
}
```

```
else
{ # we are done, return final formula, model, and statistical summary
list( formula=model.formula, model=m, multcomp.summary=lhs )
}
}
else
{
if ( length( quadratic ) > 0 ) # test for quadratic terms
{
# extract p values corresponding to quadratic terms
p <- lhs$test$pvalues[(length(terms)-length(quadratic)+1):length(terms)]
i <- which.max(p) # find the term with the smallest p values
if ( p[i] >= p.cutoff ) # is it larger than the cutoff?
{
# yes, remove term
cat("\nRemoving term", quadratic[i], "with p value", p[i], ">=", p.cutoff, "\n" )
quadratic <- quadratic[-i]
# now, recursively remove other terms
fixed.model.selection( data, response, keep.always, linear, quadratic, cubic, random, p.cutoff )
}
else
{ # we are done, return final formula, model, and statistical summary
list( formula=model.formula, model=m, multcomp.summary=lhs )
}
}
else
{
if ( length( linear ) > 0 ) # test for linear terms
{
# extract p values corresponding to linear terms
p <- lhs$test$pvalues[(length(terms)-length(linear)+1):length(terms)]
i <- which.max(p) # find the term with the smallest p values
if ( p[i] >= p.cutoff ) # is it larger than the cutoff?
{
# yes, remove term
cat("\nRemoving term", linear[i], "with p value", p[i], ">=", p.cutoff, "\n" )
linear <- linear[-i]
# now, recursively remove other terms
fixed.model.selection( data, response, keep.always, linear, quadratic, cubic, random, p.cutoff )
}
else
{ # we are done, return final formula, model, and statistical summary
list( formula=model.formula, model=m, multcomp.summary=lhs )
}
}
else
{
# we are done, return final formula, model, and statistical summary
list( formula=model.formula, model=m, multcomp.summary=lhs )
}
}
}
}


if (T)
```

```
{ quadratic=c('agg:acc','agg:ent', 'acc:ent')
      cubic=c('agg:ent:acc')
      AA <- c()
      fixedintercept <- c()
      fixedintercept.p <- c()
      acc <- c()
    acc.p <- c()
      agg <- c()
      agg.p <- c()
      ent <- c()
      ent.p <- c()
      RIs <- c()
      Raccs <- c()
      Raggs <- c()
      Rents <- c()

      aminoacids <-  c('A', 'C', 'D', 'E', 'F', 'G', 'H', 'I','K',
      'L', 'N', 'P', 'Q', 'R', 'S', 'T', 'V','Y')

      ### for E.coli, there is no K
      if (species == 'ecoli')
     { aminoacids <- c('A', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'L', 'N',
     'P', 'Q', 'R', 'S', 'T', 'V','Y')}

for (d in aminoacids )
      {
cat( "\n\n\t=== Amino acid", d, "===\n\n" )
    expr = parse( text = paste( "random.effect.selection(data = aa",d," )", sep='' ) )
r = eval(expr)
    print(r)
expr = parse( text = paste( "fixed.model.selection( data=aa",d,",response= 'optcodon', keep.always=r$keep,
linear=r$linear, quadratic=quadratic,cubic=cubic, random=r$random, p.cutoff=.05 )", sep='' ) )
      f= eval(expr)
      # extract results
 AA = c(AA, paste("aa", d, sep=''))
 ifacc_f <- F
 ifagg_f <- F
   ifent_f <- F
 accf_i <- NA
 aggf_i <- NA
   entf_i <- NA
   ifacc_r <- F
 ifagg_r <- F
   ifent_r <- F
 accr_j <- NA
 aggr_j <- NA
   entr_j <- NA

# fixed effect extraction
# only fixed intercept
 if (length(f$multcomp.summary)==0)
    {
       acc <- c(acc, NA)
      acc.p <- c(acc.p, NA)
        agg <- c(agg, NA)
        agg.p <- c(agg.p, NA)
```

```
        ent <- c(ent,NA)
        ent.p <- c(ent.p, NA)
      }
else{
for (i in 2: length(fixef(f$model)))
   {
  if (attributes(fixef(f$model))$names[[i]]=='acc')
      {
ifacc_f <- T
  accf_i <- i }
  else {ifacc_f <- ifacc_f
accf_i <- accf_i
}

  if (attributes(fixef(f$model))$names[[i]]=='agg')
      {
ifagg_f <- T
  aggf_i <- i }
  else {ifagg_f <- ifagg_f
aggf_i <- aggf_i
}

  if (attributes(fixef(f$model))$names[[i]]=='ent')
      {
ifent_f <- T
  entf_i <- i }
  else {ifent_f <- ifent_f
entf_i <- entf_i
}
  }


if (ifacc_f)
{acc <- c(acc, fixef(f$model)[[accf_i]])
  acc.p <- c(acc.p, f$multcomp.summary$test$pvalues[[accf_i]])
  }
else{
acc <- c(acc, NA)
acc.p <- c(acc.p, NA)
}

if (ifagg_f)
{agg <- c(agg, fixef(f$model)[[aggf_i]])
   agg.p <- c(agg.p, f$multcomp.summary$test$pvalues[[aggf_i]])
   }
  else{
agg <- c(agg, NA)
  agg.p <- c(agg.p, NA)
}

if (ifent_f)
{ent<- c(ent, fixef(f$model)[[entf_i]])
  ent.p <- c(ent.p, f$multcomp.summary$test$pvalues[[entf_i]])}
else{
  ent <- c(ent,NA)
  ent.p <- c(ent.p, NA)}}
```

```
 # random extraction
 # random intercept
  RIs <- c(RIs,attributes(VarCorr(f$model)$orf)$stddev[[1]])
  RIv <- RIs *RIs
 #extract random slopes if they exist
 if (length(attributes(VarCorr(f$model)$orf)$stddev)==1)
    {
  Raccs <- c( Raccs,NA)
Raccv <- Raccs*Raccs
  Raggs <- c( Raggs,NA)
Raggv <- Raggs*Raggs
Rents <- c( Rents,NA)
Rentv <- Rents*Rents
  }
  else {
 for(j in 2:length(attributes(VarCorr(f$model)$orf)$stddev))
{if (attributes(attributes(VarCorr(f$model)$orf)$stddev)$names[[j]]=='acc')
{ ifacc_r <- T
  accr_j <- j }
  else {ifacc_r <- ifacc_r
accr_j <- accr_j
}

 if (attributes(attributes(VarCorr(f$model)$orf)$stddev)$names[[j]]=='agg')
{ ifagg_r <- T
  aggr_j <- j }
  else {ifagg_r <- ifagg_r
aggr_j <- aggr_j
}

if (attributes(attributes(VarCorr(f$model)$orf)$stddev)$names[[j]]=='ent')
      { ifent_r <- T
       entr_j <- j }
       else {ifent_r <- ifent_r
     entr_j <- entr_j
  }
    }

if(ifacc_r)
{ Raccs <- c( Raccs,attributes(VarCorr(f$model)$orf)$stddev[[accr_j]])
  Raccv <- Raccs*Raccs
  }
else{
  Raccs <- c( Raccs,NA)
Raccv <- Raccs*Raccs
}


if(ifagg_r)
{ Raggs <- c( Raggs,attributes(VarCorr(f$model)$orf)$stddev[[aggr_j]])
 Raggv <- Raggs*Raggs
  }
else{
  Raggs <- c( Raggs,NA)
Raggv <- Raggs*Raggs
```

```
}

if(ifent_r)
{ Rents <- c( Rents,attributes(VarCorr(f$model)$orf)$stddev[[entr_j]])
Rentv <- Rents*Rents
}
else{
  Rents <- c( Rents,NA)
Rentv <- Rents*Rents
}
    }

   }

     # write results into a file
  RIv <- round(RIv,4)
  RIs <- round(RIs,4)
  Rentv <-round(Rentv,4)
  Rents <-round(Rents,4)
Raggv <-round(Raggv,4)
Raggs <-round(Raggs,4)
Raccv <-round(Raccv,4)
Raccs <-round(Raccs,4)
acc <- round(acc,4)
agg <- round(agg,4)
ent <- round(ent,4)
  results <- data.frame(AA, RIv, RIs,Rentv,Rents,
  Raccv,Raccs,Raggv,Raggs,acc,acc.p,agg,agg.p,ent,ent.p)

print(results)
write.table(results, "results.txt", row.names=FALSE, sep = '\t',quote=FALSE)
  }
```

Bibliography

[1] Wilke C. O. Drummond, D. A. The evolutionary consequences of erroneous protein synthesis. *Nature Reviews Genetics*, 10:715–724, 2009.

[2] Farabaugh P. J. Kramer, E. B. The frequency of translational misreading errors in e. coli is largely determined by trna competition. *RNA*, 13:87–96, 2007.

[3] H. Akashi. Synonymous codon usage in *Drosophila melanogaster*: Natural selection and translational accuracy. *Genetics*, 136:927–935, 1994.

[4] N. Stoletzki and A. Eyre-Walker. Synonymous codon usage in *Escherichia coli*: selection for translational accuracy. *Mol. Biol. Evol.*, 24:374–381, 2007.

[5] D. A. Drummond and C. O. Wilke. Mistranslation-induced protein misfolding as a dominant constraint on coding-sequence evolution. *Cell*, 134:341–352, 2008.

[6] Weems M. Wilke C. O. Zhou, T. Translationally optimal codons associate with structurally sensitive sites in proteins. *Mol. Biol. Evol.*, 26:1571–1580, 2009.

[7] Maurer-Stroh S. Schymkowitz J. Rousseau F. Reumers, J. Protein sequences encode safeguards against aggregation. *Hum. Mutat.*, 30:431–437, 2009.

[8] Serrano-L. Schymkowitz J. W. H. Rousseau, F. How evolutionary pressure against protein aggregation shaped chaperone specificity. *J. Mol. Biol.*, 355:1037–1047, 2006.

[9] Pechmann S. Dobson C. M. Vendruscolo M. Tartaglia, G. G. Life on the edge: a link between gene expression levels and aggregation rates of human proteins. *Trends Biochem. Sci.*, 32:204–206, 2007.

[10] Vendruscolo M. Tartaglia, G. G. The zyggregator method for predicting protein aggregation propensities. *Chem. Soc. Rev.*, 37:1395–1401, 2008.

[11] G. G. Tartaglia M.Vendruscolo Y. Lee, T. Zhou and C. O. Wilke. Translationally optimal codons associate with aggregation-prone sites in proteins. *Proteomics*, un:un, in press.

[12] Edgar RC. Muscle: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Res*, 32:1772 –1797, 2004.

[13] Sander C. Kabsch W. Dictionary of protein secondarystructure: pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers*, 22:2577 – 2637, 1983.

[14] Creighton TE. *Proteins: structures and molecular properties.* New York: Freeman, 1992.

[15] Haenszel W. Mantel, N. Statistical aspects of the analysis of data from retrospective studies of disease. *J. Natl. Cancer Inst.*, 22:719–748, 1959.

[16] N. Mantel. Chi-square tests with one degree of freedom; extensions of the mantel-haenszel procedure. *J. Am. Stat. Assoc.*, 58:690–700, 1963.

[17] Hochberg Y. Benjamini, Y. Controlling the false discovery rate: A practical and powerful approach to multiple testing. *J. Royal Stat. Soc.*, B 57:289–300, 1995.

[18] Robert Nelder, John; Wedderburn. Generalized linear models. *Journal of the Royal Statistical Society*, Series A (General) 135 (3):370–384, 1972.

[19] R. Schall. Estimation in generalized linear models with random effects. *Biometrika*, 78:719–727, 1991.

[20] R. Wolfinger and M. OConnell. Generalized linear mixed models: a pseudo-likelihood approach. *J. Statist. Comput. Simulation*, 48:233–243, 1993.

[21] N.E. Breslow and D.G. Clayton. Approximate inference in generalized linear mixed models. *J. Am. Stat. Assoc.*, 88:9–25, 1993.

[22] S.W. et al. Raudenbush. Maximum likelihood for generalized linear models with nested random effects via high-order, multivariate laplace approximation. *J. Comput. Graph. Statist.*, 9:141–157, 2000.

[23] J.C. Pinheiro and E.C. Chao. Efficient laplacian and adaptive gaussian quadrature algorithms for multilevel generalized linear mixed models. *J. Comput. Graph. Statist.*, 15:58–81, 2006.

[24] W.R. et al. Gilks. *Introducing Markov chain Monte Carlo. In Markov Chain Monte Carlo in Practice pp. 119,.* Chapman and Hall, 1996.

[25] R.C. et al. Littell. *SAS for Mixed Models.* SAS Publishing, 2006.

[26] J.C. Pinheiro and D.M. Bates. *Mixed-Effects Models in S and SPLUS.* Springer, 2000.

[27] G. Rodriguez and N. Goldman. Improved estimation procedures for multilevel models with binary response: a case-study. *J. R. Stat. Soc. Ser. A Stat. Soc.*, 164:339–355, 2001.

[28] H. Goldstein and J. Rasbash. Improved approximations for multilevel models with binary responses. *J. R. Stat. Soc. Ser. A Stat. Soc.*, 159:505–513, 1996.

[29] N.E. Breslow. *Whither PQL In Proceedings of the Second Seattle Symposium in Biostatistics: Analysis of Correlated Data pp. 122.* Springer, 2004.

[30] R Development Core Team. R: A language and environment for statistical computing. *R Foundation for Statistical Computing*, ISBN: 3-900051-07-0, 2010.

[31] Benjamin M. et al. Bolker. Generalized linear mixed models: a practical guide for ecology and evolution. *Trends in Ecology and Evolution*, 24:127–135, 2009.

[32] J.B. Johnson and K.S. Omland. Model selection in ecology and evolution. *Trends Ecol. Evol.*, 19:101–108, 2004.

[33] A.M. Ellison. Bayesian inference in ecology. *Ecol. Lett.*, 7:509–520, 2004.

[34] M.J. et al. Whittingham. Why do we still use stepwise modeling in ecology and behaviour? *J. Anim. Ecol.*, 75:1182–1189, 2006.

[35] K.P. Burnham and D.R. Anderson. *Model Selection and Multimodel Inference: A Practical Information-Theoretic Approach*. Springer-Verlag, 2002.

[36] P.A. et al. Stephens. Information theory and hypothesis testing: a call for pluralism. *J. Appl. Ecol.*, 42:4–12, 2005.

[37] S Greven. *Non-Standard Problems in Inference for Additive and Linear Mixed Models*. Cuvillier Verlag, 2008.

[38] A. et al. Dominicus. Likelihood ratio tests in behavioral genetics: problems and solutions. *Behav. Genet.*, 36:331–340, 2006.

[39] Walker Saveliev Zuur, Ieno and Smith. *Mixed Effects Models and Extensions in Ecology with R*. Springer, 2009.

[40] Julian J. Faraway. *Linear model with R.* Chapman & Hall/CRC, 2005.

Vita

Shujuan Feng was born in Changchun, Jilin Province, China on June 23rd, 1977. She received a Bachelor degree and Master's in chemistry from the Beijing Normal University in China in 2000 and 2003. After graduation, Shujuan taught chemistry in the Chinese Medicine School of Changchun in China. She entered the Graduate School of the University of Texas at Austin in 2007.

Permanent address: 2543 Westwind Dr
Soddy Daisy, TN 37379

This report was typeset with LaTeX[†] by the author.

[†]LaTeX is a document preparation system developed by Leslie Lamport as a special version of Donald Knuth's TeX Program.